MIKRO BITTI -kirjat

COMMODORE

HUVIA JA HYÖTYÄ 64

- Perusteellinen peliohjelmoinnin käsikirja
- 10 valmista peliohjelmaa
- 6 hyöty- ja apuohjelmaa

TARKASTAJA

Automaattinen oikolukuohjelma listausten sisäänsyöttöä varten.

Jyrki J. J. Kasvi

HUVIA JA HYÖTYÄ COMMODORE 64

Koonnut Jyrki J. J. Kasvi

BITTI -kirjat
Tecnopress Oy

HUVIA JA HYÖTYÄ COMMODORE 64

Toimittanut: Jyrki J. J. Kasvi

Ääniosuus: Juha Määttänen
Ohjelmat:
Teemu Stenberg, Harri Holopainen, Vesa Takkinen, Jaakko Leino, Juha Mikkola, Pertti Kasanen, Kristian Rautiainen, Timo Määränen, Mikko Nykyri, Antti Raudaskoski, Jukka Heino, Panu Haaramo, Tommi Timonen
Kansi ja piirrokset: Wallu
Kirjassa olevat ohjelmat on tarkastettu huolella. Emme kuitenkaan voi taata niiden virheettömyyttä emmekä vastaa esiintyneistä virheistä.

© Copyright Tecnopress Oy
Kaikki oikeudet pidätetään.

Tämän teoksen tekstiosuuden osittainenkin kopioiminen on tekijänoikeuslain (404/61, muut. 897/80) ja valokuvalain (405/61, muut. 898/80) mukaisesti kielletty ilman kustantajan kirjallista suostumusta lukuun ottamatta tätä teosta koskevia arvosteluja ja katsauksia.

Toinen tarkastettu painos
Kirjapaino Lars Eriksen Oy
Espoo 1986

ISBN 951-832-005-5

Sisällysluettelo

JOHDANTO 8

1 MITEN TEEN TOIMINTAPELIN 9
Harkintaa ja pohdiskelua 10
Rautalankamalli vuokaaviosta 12
Ongelman paloittelu 14

2 TAUSTAT 15
Merkkigrafiikka 15
Tarkkuusgrafiikka taustana 18
Grafiikkamuistin järjestely 19
Tarkkuusgrafiikan värit 20
Teoria vs. käytäntö 21
Bittiblokkina ladattava tarkuusgrafiikkaruutu 22

3 NÄYTÖN VIERITYS 23
Piste kerrallaan 23
Vieritys pidemmälle 24
Toteutus käytännössä 24
Reunat piiloon 25
Kertaus 26

4 LIIKKUVAT MERKIT 29
Merkkimuistialueen valinta 29
Merkistön siirto 30
Merkin tietojen laskeminen 31
Merkkihahmojen liikuttaminen 32

5 SPRITET 35
Spritejen näytölläolo 35
Sprite-lohko 36
Spriten sijainti näytöllä 38

Sisällysluettelo

Muita sprite-rekistereitä 39
Moniväriset spritet 43

6 ÄÄNITEHOSTEET **45**
SID-piirin ohjaus 46
Äänen korkeus 46
Verhokäyrä eli ADSR 47
Aaltomuodot 49
Äänen voimakkuus 50
Suodattimet 50
Suodatintyyppejä 51
Äänitehosteet 53
SID-rekisterikartta 56

7 OHJAUS JOYSTICKILLÄ **58**

8 PELIOHJELMAT **61**
Lunar Adventure 61
Katko 76
Jousiammunta 82
Formula 87
Avaruuden valloitus 94
Pong 103
Meteors 105
Mato 107
Starhunter 110
Näppäily 112

9 HYÖTYOHJELMAT **113**
Funktionäppäinten ohjelmointi 113
Tekstinkäsittelyohjelma 117
REM-killer 135
Sprite-editori 138
Tarkkuusgrafiikkapiirturi 152
CBM-64 Music 157

LIITE 1 OHJELMIEN SYÖTTÖOHJEET

Yleiset syöttöohjeet 167
Tarkistusohjelma 167
Konekielilataaja 169

LIITE 2 SUUNNITTELURUUDUKOITA

Spritensuunnitteluruudukko 174
Merkkihahmonsuunnitteluruudukko 175
Näytönsuunnitteluruudukko 176

Johdanto

Commodore 64 on Suomen yleisin kotimikro. Runsas ohjelmatarjonta ja hyvät tekniset ominaisuudet ovat tehneet siitä turvallisen valinnan mikron hankkijalle.

Kuusneloselle on myös julkaistu runsaasti kirjallisuutta sekä ulkomailla että suomeksi. Kuitenkin pelejä itse Basicillä ohjelmoivalle on jäänyt paljon vastaamattomia kysymyksiä.

Huvia ja hyötyä Commodore 64 vastaa yleisimpiin näistä kysymyksistä. Tarkoituksena ei ole opettaa Basicin alkeita, vaan kertoa, miten tietyt pelien peruselementit saadaan Kuusnelosella näppärästi toteutettua.

Teorian tukemiseksi on mukaan otettu myös runsaasti ohjelmalistauksia. Niitä tutkimalla ja Kuusneloselle kirjoittamalla voi tutustua ohjelmoinnin periaatteisiin käytännössä.

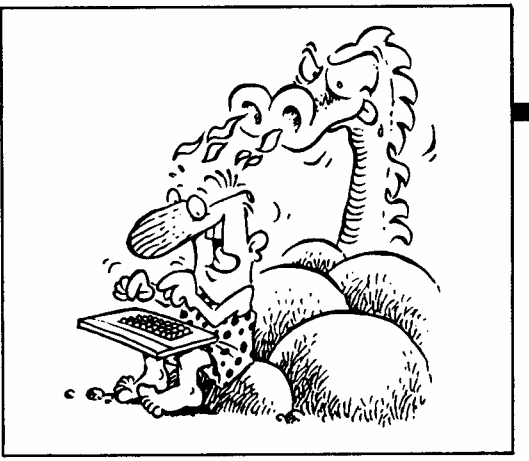
Basic-ohjelmat on listattu ohjelmalla, joka tuottaa joka ohjelmarivistä tarkistussumman. Niiden ja kirjassa olevan Tarkastaja-ohjelman avulla näppäilyvirheiden korjaaminen on helppoa, sillä joka riviltä tarvitsee tarkistaa vain tarkistussumma.

Kirjassa olevia konekieliohjelmia kirjoitettaessa tulisi käyttää niin ikään mukana olevaa konekielilataajaa, joka helpottaa pitkien konekieliohjelmien muistiinsyöttöä.

Tässä yhteydessä haluaisin myös kiittää kaikkia niitä, jotka ovat lähettäneet ohjelmiaan MikroBITTI-lehteen julkaistaviksi. Ilman heitä tämän kirjan tekeminen ei olisi ollut mahdollista.

Helsingissä 25. lokakuuta 1985

Jyrki J. J. Kasvi



1 Miten teen toimintapelin

Kun ohjelmoidaan videopelejä on aina muistettava pelaajaa, on ajateltava hänen reaktioitaan ja pyrittävä luomaan hänelle tarve pelata peliä yhä uudelleen. Tarvitaan juoni, joka antaa pelaajalle syyn tarttua peliohjaimen kahvaan.

Ensinnäkin pelissä on oltava jokin selkeä tavoite. Toiseksi tavoitteeseen on saumattomasti liityttävä jokin este, joka tekee tavoitteen saavuttamisen hankalaksi, ellei peräti vaaralliseksi.

Vastuksia kannattaa myös olla mahdollisimman monta eri tasoa, jotta mielenkiinto pelin edetessä ja taitojen karttuessa säilyy. Myös vastusten voittamisesta saatava palkkio, pisteet, tulisi sovittaa osoitetun taituruuden mukaan.

Kaikenkaikkiaan kannattaa kuitenkin muistaa, että yksinkertainen on kaunista, eli ei tulisi yrittää toteuttaa turhan monimutkaisia juonirakennelmia, jotka vain sekoittavat pelaajan pään.

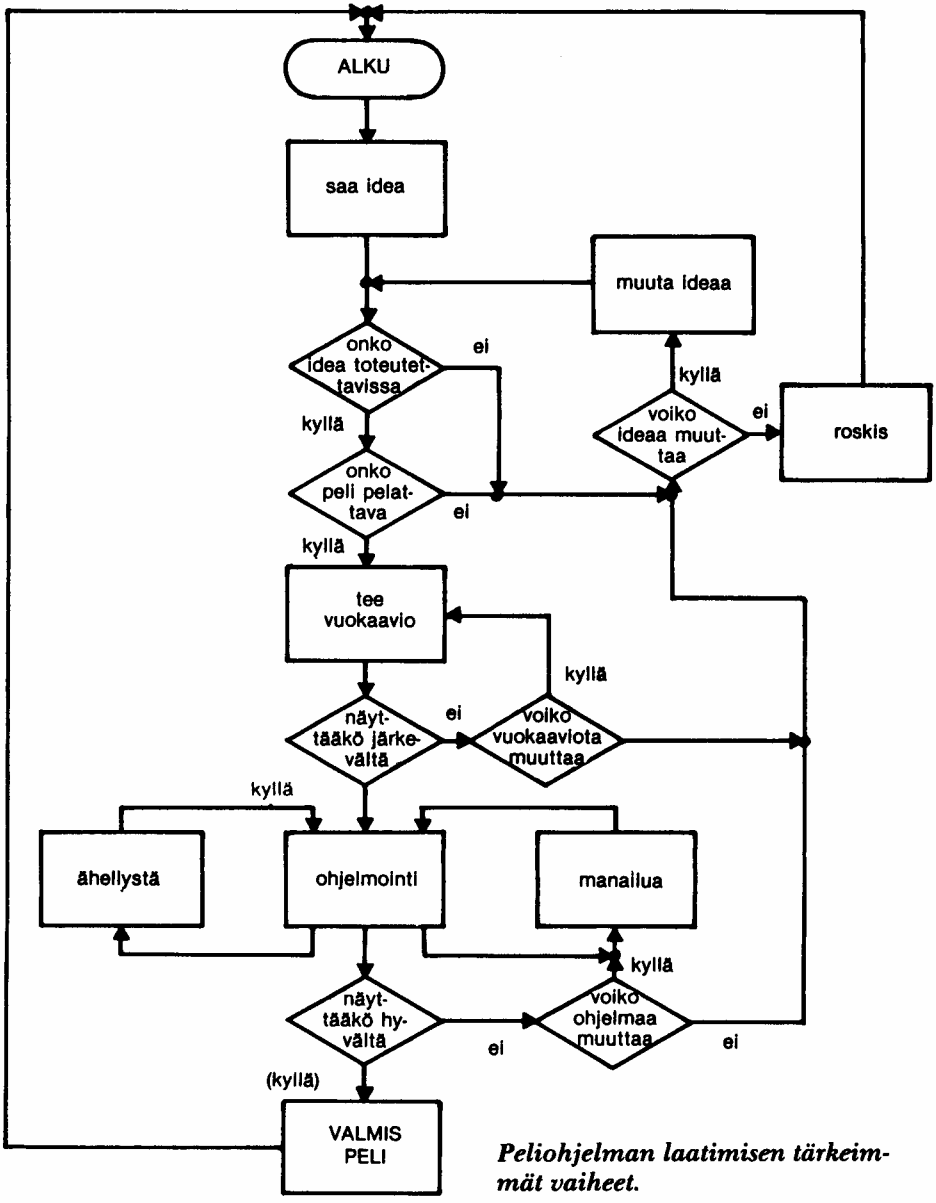
Ohjelman peruselementit määräävän juonen sommitteluun kannattaa kuluttaa runsaasti aikaa. Tämä onkin itse asiassa koko ohjelmointityön vaativin vaihe. Tällöin tehty virhe näkyy aina lopputuloksessa, eikä sitä voi mitenkään korjata tekemättä koko hommaa uudemman kerran.

Heti hyvän idean saatuaankaan ei vielä pitäisi sormet sauhuten ryhtyä

iskemään koodia koneeseen, vaan tulisi pohtia, miten hyvin ajateltu pelisommitelma olisi toteutettavissa Kuusnelosella. Viekö ohjelma liian paljon muistitilaa? Riittääkö Basicin nopeus, vai pitääkö turvautua konekieleen? Pystytäänkö pelin vaatima tulostus järkevästi toteuttamaan Kuusnelosen grafiikalla?
Mikäli vastaus yhdenkin kerran on kielteinen, ei auta muu kuin palata ideariiheen pohtimaan, voidaanko alkuperäistä ideaa sopivasti muuttaa pelattavuuden siitä kuitenkaan suuremmin kärsimättä.

Harkintaa ja pohdiskelua

Kun tai jos idea vaikuttaa toteutuskelpoiselta, voidaan siirtyä koko ohjelmointityön tärkeimpään vaiheeseen — ohjelman suunnitteluun. Mikäli ohjelman suunnittelun jättää väliin ja alkaa suoraan näpytellä ohjelmaa koneeseen, on useimmiten tuloksena muutama kilo sekavaa koodia, jota on hyvin vaikea korjata tai muuttaa. Ohjelman suoritusnopeus ei myöskään ole näin paras mahdollinen, vaikka ohjelma kaikesta huolimatta yökausien korjailun jälkeen sattuisikin toimimaan.
Aivan ensiksi kannattaa vain pohtia ohjelman eri alueita. Miten ne saa parhaiten toteutettua? Onko mahdollista hyödyntää tarkkuusgrafiikkaa? spritejä? äänitehosteita? Onko kaikki otettu huomioon? Pitäisikö jostakin tinkiä? Riittääkö Basicin nopeus? Samalla voi suunnitella ohjelman tulosteita sekä miettiä, miten käyttäjän ohjeet saadaan helpoiten kerrottua tietokoneelle.
Tämän jälkeen voikin ryhtyä kirjoittamaan paperille ongelman ratkaisumallia eli algoritmia. Ensiksi ei pitäisi suinkaan vielä käyttää Basicin käskyjä, vaan silkkaa suomen kieltä, jolla kertoo vaihe vaiheelta, miten ohjelma pääpiirteissään etenee. Toisaalta voi myös käyttää vuokaaviota ohjelman suunnittelun apuna.

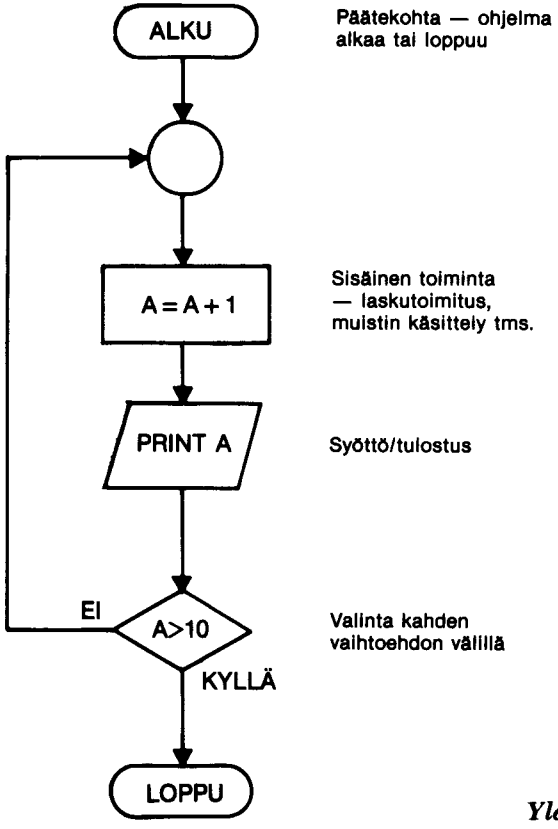


Peliohjelman laatimisen tärkeimmät vaiheet.

1 Miten teen toimintapelin

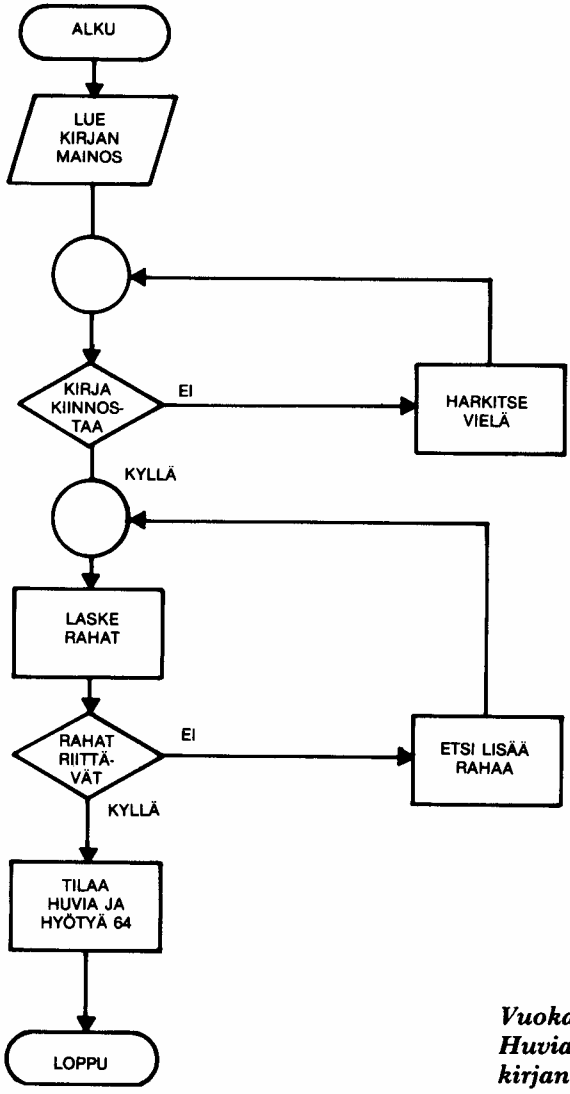
Rautalankamalli vuokaaviosta

Vuokaavio on näppärä ja havainnollinen tapa esittää jonkin ongelman ratkaisu. Tämä ei rajoitu pelkästään tietojenkäsittelyyn ja usein esimerkkeinä käytetäänkin arkielämän tavanomaisia tapahtumia.
Vuokaavio koostuu laatikoista, joiden sisälle on kirjoitettu laatikkoa vastaava toiminto, sekä näitä laatikoita yhdistävistä nuolista. Nämä nuolet kertovat sen, missä järjestyksessä laatikkojen esittämät toiminnot tulisi suorittaa. Eri tyyppisiä toimintoja kuvataan erilaisilla laatikoilla. Näistä yleisimmät on esitelty seuraavassa kuvassa.



Yleisimmät vuokaaviolaatikot.

Kokemuksen kartuttua vuokaavio on helppo muuntaa toimivaksi ohjelmaksi, jonka mahdolliset ongelmakohdat on jo etukäteen kartoitettu.

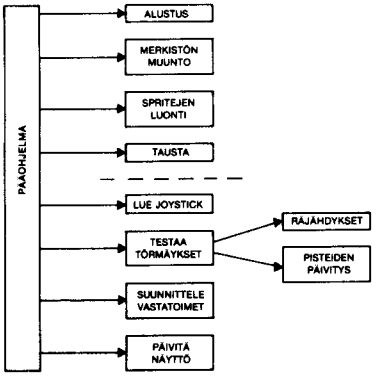


Vuokaavio siitä, miten tullaan Huvia ja hyötyä Commodore 64 -kirjan onnelliseksi omistajaksi.

1 Miten teen toimintapelin

Ongelman paloittelu

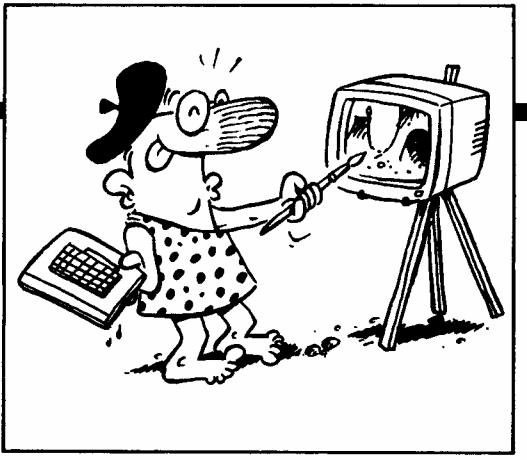
Ohjelman suunnittelua ja itse ohjelmointityötä helpottaa suuresti, jos heti alkuvaiheessa jakaa ratkaistavan ongelman selkeisiin osasiin. Näihin yksittäisiin ongelmanosiin on paljon helpompi keskittyä yksi kerrallaan kuin yrittää käsitellä koko ohjelmaa yhtenä kokonaisuutena.
Joskus näitä palasia täytyy paloitella vielä edelleen, jotta päästäisiin sopivan yksinkertaiselle tasolle. Tarpeeksi syvälle mentäessä päästään yksittäisiin ohjelmointikielen lauseisiin asti. Useissa ohjelmointikielissä koko ohjelmoinnin idea perustuu tällaiselle ongelmien jaottelulle.
Ohjelmointivaiheessa ongelmat ratkaistaan sitten yksi kerrallaan, minkä jälkeen näin syntyneet osaratkaisut (aliohjelmat) liitetään yhteen kutsumalla niitä peräjälkeen niin kutsutusta pääohjelmasta.



Eräs rakenteisesti ohjelmoidun toimintapelin malli.

Kun ohjelman suunnittelu perustuu jaottelulle tulee itse ohjelmasta hyvin selkeä, koska se jakautuu hyppykäskyistä koostuvaan lyhyehköön pääohjelmaan ja aliohjelmiin, joihin pääohjelmasta hypätään. Pääohjelmaa tutkimalla saa kerralla hyvän kuvan koko ohjelman toiminnasta ilman että täytyy tutustua yksittäisiin komentoihin. Hyvä kommentointi myös helpottaa ohjelman logiikan ymmärtämistä.
Ongelmana GOSUB-lauseista koostuvassa pääohjelmassa on ohjelman suoritusnopeuden hidastuminen lukuisten hyppyjen seurauksena, millä saattaa olla ratkaiseva merkitys lopputuloksessa kun tehdään nopeatempoista videopeliä.

2 Taustat



Usein näkee idealtaan aivan kohtalaisia pelejä, joissa koko homman pilaa huono ja ikävystyttävä tausta. Tausta määrää pelin värikkyyden ja on muutenkin ensiarvoisen tärkeä pelin selkeyden ja pelattavuuden kannalta.

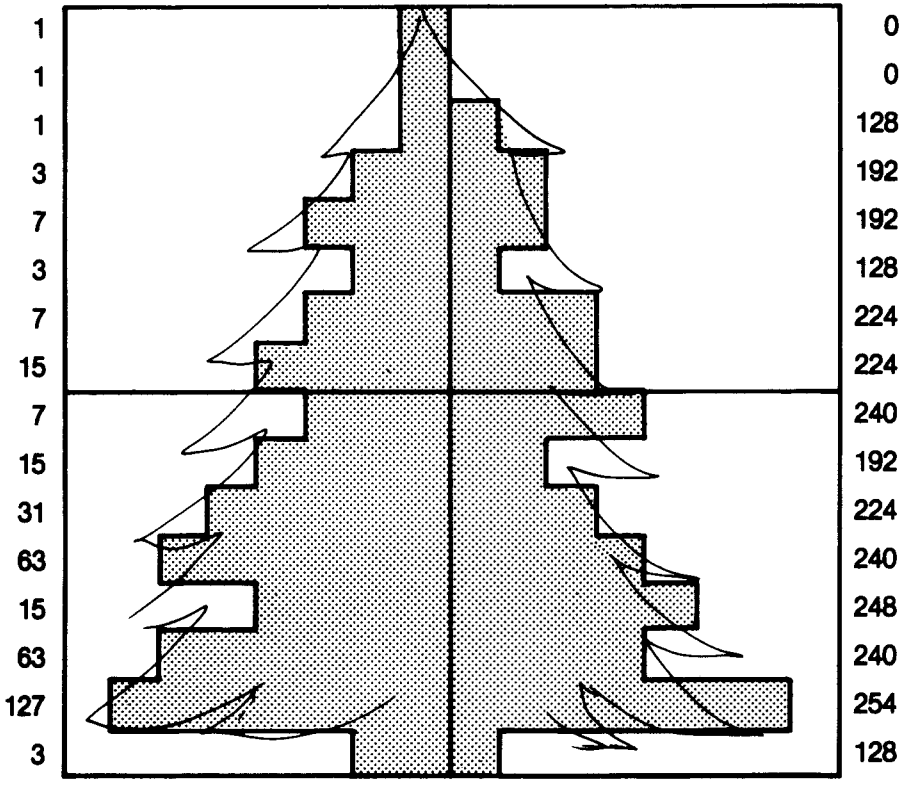
Merkkitausta

Helpoin ja yleisin tapa luoda tausta on tehdä se PRINT-lauseilla. Kuvaruutuun vain printataan 25 riviä, jotka muodostavat pelin taustan. Esimerkiksi Lunar Adventure -peli käyttää tätä menetelmää. Kun menetellään tällä tavoin voidaan hyödyntää Commodoren omia grafiikkamerkkejä, tai sitten käyttää itse määriteltyjä merkistöjä, joista enemmän luvussa 4.

Useimmissa kaupallisissa peleissä taustat on tehty käyttämällä ohjelmallisesti määriteltyjä merkkejä.

Kun käytetään omia merkkejä voidaan taustan elementtejä muodostaa useista omista merkeistä, jotka sitten printataan vierekkäin. Oheinen esimerkki näyttää, miten neljän itse määritellyn merkin avulla voidaan luoda 16X16:n pisteen kuusi.

2 Taustat



Neljästä merkistä muodostettu kuusi.

Taustaa suunniteltaessa kannattaa ensiksi piirtää taustan hahmotelma 40X25 ruutupaperille. Hahmotelman perusteella sitten sommitellaan taustan tarvitsemat ohjelmallisesti määriteltävät merkit kuin myös laaditaan taustan tulostavat 25 PRINT-lausetta.

Sokkeloita tehtäessä kulkureitille on hyvä printata välilyöntejä, jolloin on helppo tarkistaa, onko sokkelossa liikkuva sprite törmännyt sokkelon seiniin.

Parhaiten tämän menetelmän salat käyvät ilmi seuraavasta aliohjelmasta

```
8010 PRINT "<CLR/HOME><RED>#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$ <BLK>SCORE:":REM FF
8020 PRINT "<GREY3>%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&":REM E6
8030 PRINT"<RED>#$<28SPACE>#$":REM 46
8040 PRINT"<GREY3>%&<28SPACE>%& <BLK>BONUS:":REM 1F
8050 PRINT"<RED>#$<28SPACE>#$":REM 48
8060 PRINT"<GREY3>%&<4SPACE><RED> #$<14SPACE>#$<4SPACE><GREY3> %&   <GRN>▒":REM D9
8070 PRINT"<RED>#$<4SPACE><GREY3> %&<4SPACE><RED>  #$<4SPACE><GREY3>  %&<4SPACE><RED> #$  <GRN>▒":REM 59
8080 PRINT"<GREY3>%&<13SPACE>%&<13SPACE>%&   <GRN>▒":REM E1
8090 PRINT"<RED>#$<4SPACE><RED> #$<14SPACE>#$<5SPACE>#$   <GRN>▒":REM BA
8100 PRINT"<GREY3>%&<5SPACE>%&<14SPACE>%&<5SPACE>%&   <GRN>▒":REM 25
8110 PRINT"<RED>#$<28SPACE>#$   <GRN>▒":REM 09
8120 PRINT"<GREY3>%&<13SPACE><RED>#$<13SPACE><GREY3>%&   <GRN>▒":REM 8F
8130 PRINT"<RED>#$<13SPACE><GREY3>%&<13SPACE><RED>#$   <GRN>▒":REM 0D
8140 PRINT"<GREY3>%&<28SPACE>%&   <GRN>▒":REM 93
8150 PRINT"<RED>#$<4SPACE><RED> #$<14SPACE>#$<5SPACE>#$   <GRN>▒":REM B7
8160 PRINT"<GREY3>%&<5SPACE>%&<14SPACE>%&<5SPACE>%&   <GRN>▒":REM 2B
8170 PRINT"<RED>#$<13SPACE>#$<13SPACE>#$   <RED>▒":REM 54
8180 PRINT"<GREY3>%&<4SPACE><RED> #$<6SPACE><GREY3>%&<6SPACE><RED>#$<5SPACE><GREY3>%&   <RED>▒":REM DC
8190 PRINT"<RED>#$<5SPACE><GREY3>%&<14SPACE>%&<5SPACE><RED>#$   <RED>▒":REM 5C
```

```
8200 PRINT"<GREY3>%&<28SPACE>%&    <RED>▒":REM
  8E
8210 PRINT"<RED>#$<28SPACE>#$   ":REM 46
8220 PRINT"<GREY3>%&<28SPACE>%& <BLK>FUEL":REM
   8A
8230 PRINT "<RED>#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$#$
  $#$":REM 2A
8240 PRINT "<GREY3>%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&%&
  &%&%& <BLK>. . .":REM 04
```

Tarkkuusgrafiikka taustana

Commodoren näyttö jakautuu tarkkuusgrafiikkatilassa 200×320 pisteeseen. Kun halutaan asettaa Kuusnelonen tarkkuusgrafiikkatilaan tulee muistipaikan 53265 viitos-bitin arvo asettaa ykköseksi seuraavasti:

POKE 53265, PEEK(53265) OR (2↑5)

Takaisin tekstitilaan päästään vastaavasti nollaamalla viitos-bitti:

POKE 53265, PEEK(53265) AND (255-2↑5)

Tämän jälkeen ei kuitenkaan vielä tiedetä, mistä saadaan tieto siitä, mikä piste tarkkuusgrafiikkaruudussa on sytytetty ja mikä ei. Täytyykin siis määritellä muistialue, josta tietokone hakee tarvittavan tiedon. Yleensä käytetään aluetta 8192 - 16191, jonka käyttö tähän tarkoitukseen kerrotaan tietokoneelle asettamalla muistipaikan 53272 kolmos-bitti ykköseksi, eli

POKE 53272, PEEK(53272) OR (2↑3)

Nyt voi hyvin kokeilla muistipaikan 53265 5-bitin ja muistipaikan 53272 3-bitin asettamista ykköseksi. Tuloksena lienee kuvaruudullinen roinaa, ellei sitten ole juuri ennen pelannut jotakin tarkkuusgrafiikkaa hyödyntävää peliä, jonka tarkkuusgrafiikkakuvat vielä sattuvat olemaan muistissa. Joka tapauksessa jos halutaan tehdä tarkkuusgrafiikkatilassa jotakin tulee grafiikkamuisti tyhjentää, eli kaikkiin alueen muistipaikkoihin tulisi kirjoittaa nolla. Tämä käy seuraavasti:

FOR SI=8192 TO 16191: POKE SI,0: NEXT SI

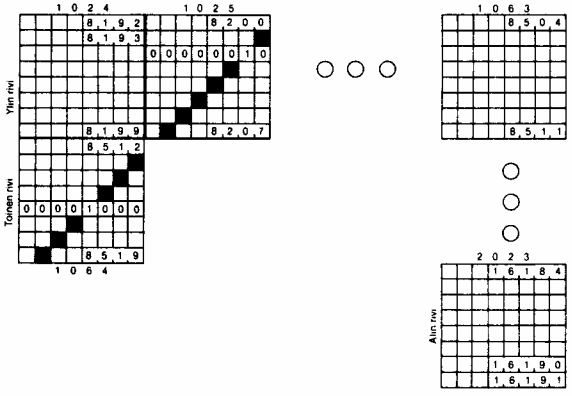
Grafiikkamuistin järjestely

Se, miten tieto grafiikkanäytön pisteiden tilasta käytetylle muistialueelle on tallennettava, onkin jo vähän monimutkaisempi juttu.

Yhden kahdeksanbittisen tavun sisältävä muistipaikka pitää sisällään tiedon kahdeksan pisteen tilasta. Eli jos vain tavun alin bitti on yksi (00000001), on sitä näytöllä vastaava piste sytytetty ja muita bittejä vastaavat pisteet sammutettu.

Toisaalta muistipaikkojen järjestys ei suinkaan suoraan vastaa näytön pisteiden järjestystä. Muistipaikka 8192 toki kertoo ylärivin kahdeksan ensimmäisen grafiikkapisteen tilan, mutta jo 8193 kertookin puolestaan toisen rivin kahdeksan ensimmäisen pisteen tilan. Samaan tapaan jatketaan muistipaikkaan 8199, joka kertoo kahdeksannen rivin kahdeksan ensimmäisen pisteen tilan, mutta muistipaikka 8200 kertookin ensimmäisen rivin pisteiden 9 - 16 tilan.

Itse asiassa grafiikkamuisti sisältää tiedon grafiikkapisteiden tilasta aina yhden merkin alueelta kerrallaan. Kun ensimmäinen merkkirivi on käyty läpi, sama toistuu seuraavan rivin osalta jne. aina grafiikkanäytön oikeaan alalaitaan asti. Oheinen kuva selvittää käytettyä järjestelyä.



Yksi muistipaikka (8192-16191) sisältää tiedon kahdeksan grafiikkapisteen tilasta. Esimerkiksi muistipaikan 8516 sisältönä olisi tässä tapauksessa 2↑3=8.

Yksi muistipaikka (1024-2023) sisältää tiedon pisteiden ja taustan väreistä aina yhden merkin alueelta (8×8 pistettä).

Muistipaikkojen ja grafiikkapisteiden sijainnin välisestä ristiriidasta seuraa se, että on varsin hankalaa selvittää, miten määritellä koordinaatteja X ja Y vastaava muistipaikan osoite ja vielä tästä muistipaikasta muutettava bitti. Ohessa on kuitenkin esitetty aliohjelma, joka sytyttää pisteen, jonka koordinaatit vasemmasta yläkulmasta luettuna ovat vaakasuuntaan X ja pystysuuntaan Y. Grafiikkamuistin alkuosoite on sijoitettu muuttujaan GA (GA=8192).

```
READY.

1000 X1=INT(X/8): Y1=INT(Y/8): AP=YAND7
1010 G=GA+X1*8+Y1*320+AP: PS=7-(XAND7)
1020 POKE G,PEEK(G) ORPS
1030 RETURN
```

Kun halutaan sammuttaa piste X,Y, tulisi POKE-lauseen kuulua seuraavasti:

POKE G, PEEK(G) AND (255-PS)

Tarkkuusgrafiikan värit

Grafiikkapisteiden ja taustan värit voidaan määrätä aina yhden merkin alueella. Tämä tieto on talletettu normaalitilassa näyttömuistina käytetylle alueelle (1024 - 2023). Näin ollen näyttömuistikin täytyy tyhjentää ennen grafiikkatilan käyttöä lauseella

PRINT CHR$(147)

ellei sitten tarkoituksella halua ohjata grafiikkanäytön värejä printtaamalla näyttömuistiin jotakin.

Kukin näyttömuistin muistipaikka ohjaa vastaavan näytön merkkipaikan alueella olevien 64 pisteen värejä. Muistipaikan merkitsevimmät neljä bittiä (vasen puolisko) määräävät sytytettyjen pisteiden ja oikea puolisko sammutettujen pisteiden eli taustan oheisen taulukon mukaisesti. Eli jos muistipaikan 1024 sisältönä on 00110100 (=52), ovat äärimmäisenä vasemmassa yläkulmassa olevan merkin alueella sytytettyinä olevat pisteet sinivihreitä, ja tausta on purppuranpunainen.

Arvo	Vastaava väri	Arvo	Vastaava väri
0	Musta	8	Oranssi
1	Valkoinen	9	Ruskea
2	Punainen	10	Vaaleanpunainen
3	Syaani	11	Tummanharmaa
4	Purppura	12	Harmaa
5	Vihreä	13	Vaaleanvihreä
6	Sininen	14	Vaaleansininen
7	Keltainen	15	Vaaleanharmaa

Seuraava aliohjelma värjää sen merkkipaikan, jossa piste X,Y on, sytytetyt pisteet muuttujan VA arvon mukaisesti ja sammutetut pisteet muuttujan TA arvon mukaisesti.

```
2000 X1=INT(X/8): Y1=INT(Y/8)
2010 G=1024+X1+40*Y1: SI=16*VA+TA
2020 POKE G,SI: RETURN
```

On siis todellakin muistettava, että pisteen värin muuttaminen muuttaa myös kaikkien saman merkin alueella olevien pisteiden värin. Sama pätee taustaan.

Teoria vs. käytäntö

Käytännössä ohjelmallista tarkkuusgrafiikkaa on erittäin työlästä hyödyntää pelien taustoissa. Tulos on kyseenalainen ja toteutus hidasta. Toisaalta tarkkuusgrafiikkaa voi käyttää luomalla ensin erityisellä grafiikkaohjelmalla pelille hyvä taustakuva, jonka voi ladata erikseen massamuistilaitteelta muistiin ns. bittiblokkina.

Bittiblokkina ladattava tarkkuusgrafiikkaruutu

KoalaPadilla ja muilla grafiikkaohjelmilla tehtyjä tarkkuusgrafiikkakuvia voidaan myös käyttää pelien taustana tai erityisesti pelien latauskuvina, joita pidetään näytössä latauksen ajan.

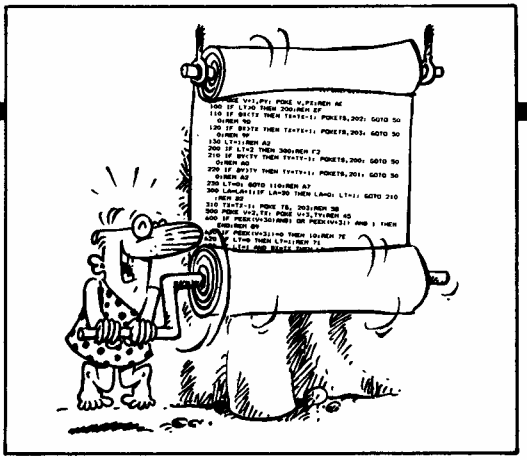
Käytetyt tallennusmuodot vaihtelevat. Onko tieto tallennettu numeroina vai merkkeinä? Miten näyttömuistiin tulevat värit ja grafiikkamuistiin tulevat tiedot grafiikkapisteiden tilasta on tallennettu toistensa suhteen? Kuitenkin kun tämä tiedetään, voidaan väri- ja grafiikkatiedot ladata massamuistilaitteilta suoraan paikoilleen ohjelmasta.

Jos halutaan pitää etukäteen luotua kuvaa näytössä latauksen ajan, täytyy ohjelma paloitella seuraavasti:

— Latausohjelma, joka hakee latauskuvan grafiikkamuistiin ja lataa muun ohjelman sen jälkeen muistiin ja käynnistää sen.
— Aloituskuva, joka ladataan grafiikkamuistiin erityisellä latausohjelmalla.
— Varsinainen peliohjelma, jonka ensimmäiseksi ladattava latausohjelma hakee muistiin aloituskuvan jälkeen.

Seuraava esimerkkiohjelma lataa levyltä grafiikkapiirturilla luodun kuvan piirturin latauksen ajaksi. Kuva on tiedostossa nimeltä KUVA.PRG. Grafiikkapiirturiohjelman listaus on kirjan ohjelmaosassa.

```
0 POKE 53265, PEEK(53265)OR(2↑5): POKE 5
  3272,PEEK(53272)OR(2↑3):REM 88
10 OPEN 1,8,1,"KUVA,R":REM AA
20 FOR SI=8192 TO 16191:REM 9B
30 GET#1,GR$: IF GR$=""THEN30:REM 19
40 GR=ASC(GR$):POKE SI,GR:REM E9
50 NEXT: CLOSE 1:REM 85
60 PRINT "<CLR>LOAD"+CHR$(34)+"PIIRTURI"
  +CHR$(34)+",8";:REM A7
65 PRINT "<4DOWN>RUN<SUP>;":REM 85
70 POKE 631,19:POKE632,13:POKE633,13:POK
  E634,13:POKE198,4:REM F0
```



3 Näytön vieritys (scrollaus)

Kaupallisissa peleissä herättää usein ihastusta tyylikkäästi toteutettu pehmeä näytön vieritys johonkin suuntaan. Basicillä se tuntuu täysin mahdottomalta. Näin asia ei itse asiassa kuitenkaan ole, mutta Basicillä toteutettuna vierityksen vaatima muistipaikkojen käsittely vie liian paljon aikaa. Konekielellä puolestaan näytön nopea vieritys on helppo toteuttaa.

Piste kerrallaan

Kuusnelosessa on rekisteri (muistipaikka), jonka sisältöä muuttamalla saadaan näytöllä oleva teksti liukumaan piste kerrallaan johonkin suuntaan, kuitenkin enintään seitsemän pisteen verran. Näytön reunat eivät kuitenkaan siirry, vaan näytön sisältö 'liukuu' reunojen 'alla'. Tätä voi mielessään havainnollistaa kuvittelemalla, että näytön reunat ovat taulun kehykset ja ruudulla oleva teksti on taulu, jota sitten liikutellaan kiinteiden kehysten takana.

Näyttöä voidaan siirtää piste kerrallaan yhden merkin verran alas tai oikealle. Kun vieritetään näyttöä vasemmalle tai ylös, täytyy näyttö ensin

siirtää nopeasti vastakkaiseen ääriasentoonsa (piste kerrallaan siirtämisen sijasta kerralla), mistä se sitten piste kerrallaan siirretään takaisin.

Pehmeä näytön vieritys yhden merkin korkeudelta/leveydeltä tapahtuu kahdeksassa vaiheessa. Tämä toteutetaan sijoittamalla vieritysrekisteriin arvot 0-7. Alkuarvona näissä rekistereissä ei toki ole nolla, mutta sillä ei ole suurempaa merkitystä.

Vieritys seitsemää pistettä pidemmälle

Jos vierityksen tulee jatkua yhden merkin suuruisen siirtymän jälkeen, ei tehtävissä ole muuta kuin siirtää koko näyttömuistin sisältöä merkin verran haluttuun suuntaan, ja aloittaa piste kerrallaan tapahtuva näytön sisällön liu'uttaminen jälleen alusta.

Tämä juuri on näytön vierityksen hankala vaihe. Basic-kieli ei kykene tähän niin nopeasti, että scrollaus ei hidastuisi tarpeettoman paljon. Basicillä vieritys onnistuukin vain alhaalta ylös, jolloin voidaan oheisen esimerkin tapaan hyödyntää normaalia vieritystä, joka siirtää näyttömuistin sisällön yhden merkin verran ylöspäin. Näyttömuistin sisällön siirto on kuitenkin helppo toteuttaa pienellä konekielirutiinilla.

PROGRAM:RUTINABITTI

```
10 PRINT "<HOME>";CHR$(14);:REM 72
20 FOR J=1 TO 24: PRINTCHR$(13);: NEXT J:REM 85
25 POKE53265,PEEK(53265)AND247:REM B2
30 PRINT "<4SPACE>↑IKRO|↘||↘":REM 32
40 FOR Y=7TO 0 STEP -1:REM 85
50 POKE 53265, (PEEK(53265)AND248) + Y:REM 86
60 FOR VI=1 TO 50: NEXT:REM DB
70 NEXT: GOTO 30:REM 7C
```

Toteutus käytännössä

Näytön vieritystä ohjaavat rekisterit sijaitsevat VIC-II -piirissä. Vaakasuuntaan vieritettäessä onkin siis käsiteltävä muistipaikkaa 53270 ja pys-

tysuuntaan vieritettäessä muistipaikkaa 53265.

Jos esimerkiksi vieritetään näyttöä vaakasuuntaan vasemmalta oikealle, on muistipaikan 53270 kolme alinta bittiä kasvatettava nollasta seitsemäksi. Nämä kolme bittiä sisältävät tiedon näytön asemasta. Parhaiten näytön pehmeä siirto merkin verran vasemmalta oikealle käy päinsä seuraavantyyppisillä lauseilla

```
10 FOR X=0 TO 7
20 POKE 53270, (PEEK(53270) AND (255-2↑2-2↑1-2↑0)) + X
30 NEXT X
```

missä PEEK(53270) AND (255-2↑2-2↑1-2↑0) nollaa muistipaikasta 53270 saatavan luvun kolme alinta bittiä, joihin on hyvä sijoittaa luku väliltä 0-7 (+ X). Lopputulos voidaankin sitten sijoittaa takaisin muistipaikkaan 53270 (POKE 53270).

Vastaavasti menetellään kun vieritetään näyttöä ylhäältä alas sillä erolla, että nyt toimenpiteet kohdistuvat muistipaikkaan 53265.

Scrollattaessa näyttöä oikealta vasemmalle tai alhaalta ylös asianomaisen rekisterin sisältöä tietystikin supistetaan seitsemästä nollaan.

Reunat piiloon

Kuvaruudun reunat nykivät edestakaisin, mikäli scrollaus suoritetaan ilman erikoistoimenpidettä, jolla näytön reunoilta pimennetään tarpeelliset vaakarivit tai pystysarakkeet. Tällöin nykiminen jää levenneiden näytön reunojen alle piiloon, vaikka se itse asiassa jatkuu edelleen.

Usein olisi myös hyvä, jos näytön reunan takana olisi tilaa, johon voisi piilottaa näytölle vieritettävän tekstin tai omien merkkien avulla luodun kuvan osan. Tämä on mahdollista siten, että näytön reunoista rajataan kummaltakin puolen yksi pystysarake piiloon ja/tai näytön ylälaidasta rajataan vaakarivi piiloon. On muistettava, että tämä toimenpide ei vaikuta mitenkään näyttömuistiin, vaan siitä jätetään vain osa tulostamatta ruutuun, jolloin se jää levenneiden näytön reunusten alle.

Kun rajataan näytön reunoista sarakkeet pois asetetaan muistipaikan 53270 kolmosbitti nollaksi seuraavanlaisella lauseella

```
POKE 53270, PEEK(53270) AND (255-2↑3)
```

missä 255-2↑3 vastaa binäärilukua 11110111, jossa kolmosbitti on nolla

(tämän vuoksi yhtälössä kakkosen potenssina juuri on kolmonen). Vastaavasti näyttö saadaan takaisin normaaliksi asettamalla kyseinen bitti ykköseksi seuraavaan tapaan

POKE 53270, PEEK(53270) OR (2↑3)

Varsinaisissa ohjelmissa potenssit ja yhteenlaskut voidaan toki laskea jo ennalta valmiiksi (255-2↑3 = 247).

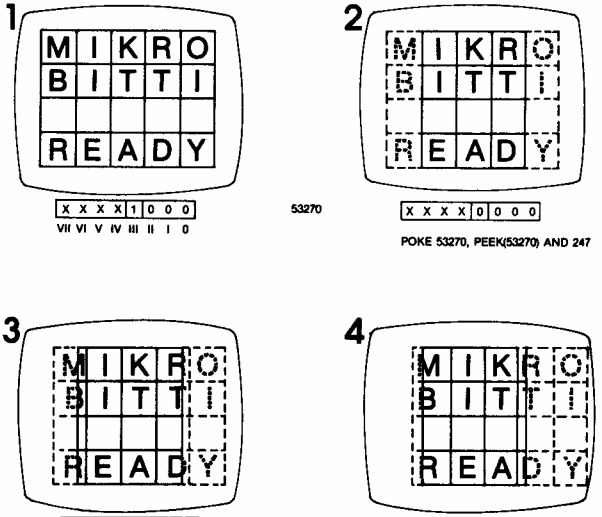
Itse asiassa kun rajataan näytön sivuilta pystysarakkeet pois menee oikeasta reunasta piiloon yksi merkkisarake + yksi piste seuraavasta merkkisarakkeesta ja vasemmasta reunasta yksi merkkisarake - yksi piste, mutta sillä ei ole suurempaa käytännön merkitystä.

Pystysuunnassa toimitaan vastaavasti muistipaikan 53265 suhteen. Tällöin kylläkin näytöltä jää piiloon vain yksi (ylimmäinen) vaakarivi.

Kertaus

Kokonaisuudessaan näytön vieritys koostuu seuraavista vaiheista:

— Poistetaan näytön reunasta rivi tai/ja sarakkeet, jotta näytön reuna ei vierityksen aikana nykisi.
— Asetetaan näytön aseman määräävä rekisteri (53265:n tai 53270:n kolme alinta bittiä) jompaankumpaan ääriarvoonsa (0 tai 7) vieritys-suunnasta riippuen.
— Pienennetään tai kasvatetaan näytön aseman määräävän rekisterin sisältö 0:sta 7:ään tai 7:stä 0:aan, jolloin näyttö siirtyy pehmeästi haluttuun suuntaan yhden merkin verran.
— Siirretään kerralla koko kuvaruutumuistin sisältö yhden merkin verran vierityksen suuntaan, jolloin äärimmäisenä vierityssuunnassa ollut rivi tai sarake poistuu muistista ja näyttömuistin toiseen laitaan jää tilaa uudelle näytölle vieritettävälle riville tai sarakkeelle.
— Sijoitetaan näytölle vieritettävät uudet merkit näytön käyttämättömälle kätketylle riville tai sarakkeelle.
— Lopuksi palataan takaisin pistekohtaiseen siirtoon (0:sta 7:ään tai 7:stä 0:aan), jotta voitaisiin jatkaa vieritystä toisten kahdeksan pisteen verran. Kun edellä vierityksessä päästiin toiseen äärilaitaan, esimerkiksi 7:ään ja vielä sen jälkeen kerralla siirrettiin näyttömuistin sisältöä merkin verran, 'kumotaan' merkkimuistin siirto POKEamalla vieritysrekisteriin 0.



Scrollaus vaakasuunnassa vaihe vaiheelta

1. Normaalitila.

2. Näytön reunoista rajataan pystysarakkeet pois nollaamalla 53270:n kolmosbitti.

3. Näyttöä siirretään reunan 'alla' yhden merkin verran oikealle kasvattamalla 53270:n kolmen alimman bitin sisältöä nollasta seitsemään.

6. Kun siirto on yhden merkin leveydeltä valmis, vieritystä jatketaan siirtämällä koko näyttömuistin sisältö yhden merkin verran oikealle.

7. Sitten 53270:n kolme alinta bittiä nollataan nopeasti, jotta näyttö näyttäisi samalta kuin ennen näyttömuistin siirtoa.

8. Tähän tyhjään pystysarakkeeseen voidaan sijoittaa uutta, näytölle jatkossa scrollattavaa tekstiä. Tämän jälkeen siirrytään takaisin ruutuun kolme.

3 Näytön vieritys

Vaikka koko näyttömuistin siirtäminen viekin kohtuuttomasti aikaa Basicillä, voidaan pienempiä näytön osia kuten yksittäisiä rivejä tai sarakkeita vierittää pehmeästi näytön poikki. Seuraavat esimerkit näyttävät, miten tämä tehdään vaakasuunnassa joko pokeilemalla näyttömuistiin siirtymän mukaisesti tai printtaamalla liu'utettava teksti aina eri paikkaan.

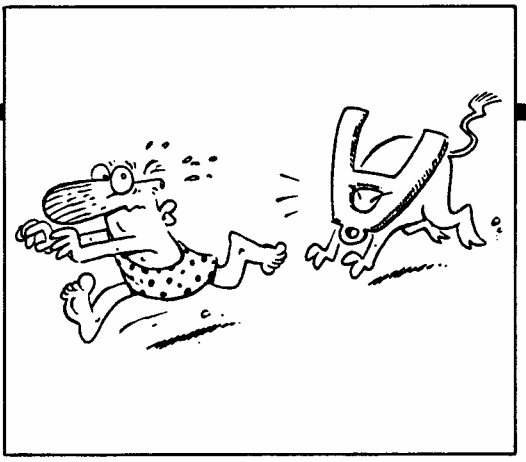
```
PROGRAM:LIUKUBITTI

10 PRINT"<CLR/HOME><GRN>";CHR$(14):REM D5
20 PRINT" ↑IKRO|↑||↑<10LEFT>";:REM 8E
30 FOR X=0 TO 7:REM E9
40 POKE 53270, (PEEK(53270) AND 248) + X:REM 7C
45 FOR VI=1 TO 5: NEXT VI:REM 4D
50 NEXT X:REM FC
60 POKE 53270, (PEEK(53270) AND 248):REM FB
70 GOTO 20:REM 02


PROGRAM:MATO

10 FOR J=1 TO 26:REM 0B
15 POKE53270,PEEK(53270)AND (255-2↑3):REM E9
20 PRINT"<BLK><40SPACE>";:REM FE
30 PRINT"<PUR><40SPACE>";:REM 0B
40 NEXT J:REM ED
50 POKE53270, PEEK(53270) AND (255-2↑3):REM E8
60 FOR PA=1304 TO 1504:REM 50
70 FOR PA=1304 TO 1504: POKE PA-1,81: POKE PA,87: POK
  E PA+39,75: POKE PA-41,73:REM 1E
80 FOR X=0 TO 7: POKE53270,(PEEK(53270)AND248)+X: FOR
   J=1 TO15: NEXT J,X:REM 69
90 POKEPA-5,32:POKE PA+35,32:POKEPA-45,32:POKE53270,P
  EEK(53270)AND248:NEXTPA:REM E3
100 POKE53270, PEEK(53270) OR (2↑3):REM 19
```

4



Liikkuvat merkit

Usein on tarpeen muuttaa Kuusnelosen vakiomerkistöä. Uutta merkistöä käytetään joko taustassa tai pelihahmoina tai sitten on vain mukavaa, jos ritarimaailmaan sijoittuvan pelin tekstit on kirjoitettu vanhahtavilla aakkosilla. Esimerkki tällaisesta merkistön käytöstä on Lunar Adventure.

Jokainen merkki koostuu 8×8 pisteen matriisista. Kukin pisteistä on joko sytytetty tai sammutettu. Näin yhden merkin vaatima tietomäärä voidaan hyvin esittää kahdeksan kahdeksan bittiä sisältävän muistipaikan avulla.

Merkkimuistialueen valinta

Mistä tietokone sitten tietää, mistä hakea merkin tulostamiseksi tarvittavat tiedot tarpeen vaatiessa? Muistipaikan 53272 neljä alinta bittiä sisältävät tiedon siitä, mistä muistin lohkosta tiedot merkistöstä kulloinkin haetaan. Tätä voi hyvin kokeilla POKE-lauseella

POKE 53272,(PEEK(53272)AND240)+12

joka käskee tietokoneen hakea tulostettavien merkkien tiedot muistipaikasta 12288 eteenpäin normaalin muistipaikan 53248 sijasta.

Esimerkin seurauksena näyttö menee kuitenkin täysin sekaisin. Tietenkin! Syynä on se, että muistipaikasta 12288 eteenpäin ei ole kirjoitettu

mitään merkkitietoja, vaan satunnaista roinaa. Näin ruudulla olevat merkitkin muuttuvat satunnaisen näköisiksi, vaikka niiden merkitys ohjelmassa säilyykin entisenlaisena.

Normaalitilaan pääsee takaisin pokeamalla samaan tapaan 53272:n neljään alimpaan bittiin viitosen (...+5). Kokeilla voi myös, mitä tapahtuu kun sinne pokeaa kuutosen. Kun palautetaan merkistöä takaisin täytyy olla varsin huolellinen, koska näytölle tulostuvat merkit ovat sattuneesta syystä ulkonäöltään täysin sekaisin.

Merkistön siirto

Vaikka haluttaisiinkin muuttaa merkistöstä vain osa, on alkuperäinen merkistö siirrettävä ensin pois ROM-muistista, missä se on muistipaikasta 53248 eteenpäin. Näin se saadaan RAM-muistiin, jolloin haluttua osaa merkistöstä voidaan hyvin muuttaa.

Ongelmana siirrossa on kuitenkin se, että tietyt syöttö- ja tulostusrekisterit itse asiassa sijaitsevat samalla muistialueella kuin alkuperäinen merkkimuisti. Kuusnelonen vaihtaa tämän muistialueen merkitystä aina tarpeen mukaan. Jos halutaan ohjelmallisesti lukea merkkimuistia, on tietokone luvun ajaksi saatava pysymään tilassa, jossa muistipaikassa 53248 todella on @-merkin ensimmäiset datat. Tämä saadaan aikaan estämällä näppäimistönlukukeskeytykset lauseella

POKE 56334, PEEK(56334) AND 254

ja käskemällä Kuusnelosta pitämään muistipaikkaa 53248 todella merkkimuistin alkuosoitteena syöttö-tulostusrekisterialueen alun sijasta. Tämä tapahtuu lauseella

POKE 1, PEEK(1) AND 251

Vastaavasti tilanne on merkkimuistin luvun jälkeen palutettava ennalleen lauseilla

POKE 56334, PEEK(56334) OR 1
POKE 1, PEEK(1) OR 4

Ennen merkistön siirtoa RAM-muistiin täytyy kuitenkin huolehtia siitä, että muistissa oleva ohjelma ei vahingossa joudu samalle muistialueelle kuin merkistö. Jos käytetään muistialuetta muistipaikasta 12288

eteenpäin, varaavat seuraavat lauseet merkistölle tarpeeksi tilaa muistista

POKE 52, 48: POKE 56, 48

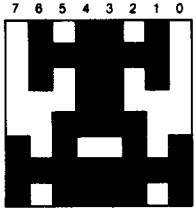
Kun merkit on saatu siirrettyä RAM-muistiin, niitä voidaan muuttaa tarpeiden mukaan. Yleisimmin tämä tehdään siirtämällä DATA-lauseista saatavat luvut uuteen merkkimuistiin. Esimerkkiohjelma muuttaa @-merkin avaruusraketiksi.

PROGRAM: RAKETTI

```
10 POKE 53272, (PEEK(53272)AND240)+12:REM 80
20 POKE56334,PEEK(56334)AND254:REM AB
30 POKE1,PEEK(1)AND251:REM 01
40 FOR SI=0 TO  511:REM 8E
50 POKE SI+12288, PEEK(SI+53248):REM CF
60 NEXT SI:REM 41
70 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1:R
   EM 56
80 FORSI=0TO7:README:POKESI+12288,ME:NEXT:REM
   86
90 PRINT"@@@@@@@@@@@@@@@@":REM 7A
100 DATA 4,4,14,14,14,31,31,27,17:REM 3B
```

Merkin datojen laskeminen

Muistiin sijoitettavat luvut lasketaan merkin määrittelevästä 8×8-matriisista oheisen kuvan esittämällä tavalla. Merkin muodostavat kahdeksasta pisteestä koostuvat rivit muutetaan ensin binaariluvuiksi siten, että piste vastaa ykköstä ja tyhjä nollaa. Nämä kahdeksan binaarilukua muutetaan sitten desimaaliluvuiksi, jotka sijoitetaan muistiin haluttua merkkiä vastaavalle paikalle.



211+213+214+216= 90
211+212+213+214+215+216= 126
211+213+214+216= 90
213+214= 24
212+213+214+215= 60
210+212+215+217= 165
210+211+212+213+214+215+216+217=255
210+212+213+214+215+217= 189

Esimerkki siitä, miten uuden merkin vaatimat datat lasketaan 8×8 mallista.

4 Liikkuvat merkit

Se muistialue, johon lasketut kahdeksan lukua sijoitetaan saadaan kun muistetaan, että jokainen merkki vaatii kahdeksan muistipaikkaa ja tiedetään se järjestys, missä merkit muistissa ovat. Tämän kertova taulukko löytyy liiteosasta (sama taulukko, joka kertoo merkkejä vastaavat POKE-arvot). Näin merkistön siirron jälkeen merkin ensimmäinen muistiosoite saadaan lauseesta

12288 = 8×JL

missä JL kertoo sen, monesko merkki on kyseessä. (@:n järjestysluku on 0.)

Merkistöä muunnettaessa on myös muistettava, että RUN/STOP ja RESTORE -näppäinten yht'aikainen painaminen palauttaa merkistön ennalleen.

Merkkihahmojen liikuttaminen

Animaation idea on periaatteessa hyvin yksinkertainen. Ensiksi piirretään liikuteltavaksi haluttu kuvio. Sitten sitä pidetään näkyvillä hetken aikaa, minkä jälkeen se poistetaan näkyvistä. Seuraavaksi sama kuvio piirretään uudelleen, tällä kertaa kuitenkin hieman eri paikkaan ja/tai hieman eri näköisenä. Kun tämä toistetaan tarpeeksi monta kertaa peräkkäin tarpeeksi nopeaan tahtiin saadaan sellainen vaikutelma, että kuvio todella liikkuu.

Tietokoneella tämä on helposti toteutettavissa. Kun halutaan piirtää kuvio se printataan näyttöön. Kun kuvio pyyhitään näkyvistä sen tilalle printataan tyhjää. Yksinkertaisimmillaan tämä on toteutettu seuraavassa ohjelmanpätkässä, jossa sama PRINT-lause sekä pyyhkii että tulostaa pallon, joka näyttää vierivän näytössä vasemmalta oikealle.

```
PROGRAM:PALLO

10 PRINT "<CLR/HOME><YEL>";:REM 9E
20 PRINT" ●<LEFT>";:REM DC
30 FOR VI= 1 TO 100: NEXT VI: REM VIIVESILMUKKA:REM A
  5
40 GOTO 20:REM FF
```

Huomattavaa on kursorinohjausmerkki tulostettavan rivin lopussa. Se siirtää tulostusta yhden merkin verran taaksepäin, jolloin seuraavalla kerralla tulostettava välilyönti todellakin tulee tällä kerralla tulostetun pallon paikalle.

Hieman monimutkaisempi tapaus on vuorossa seuraavassa, missä tyttö vilkuttaa rakastetulleen sydän pomppoillen.

PROGRAM: TYTTO

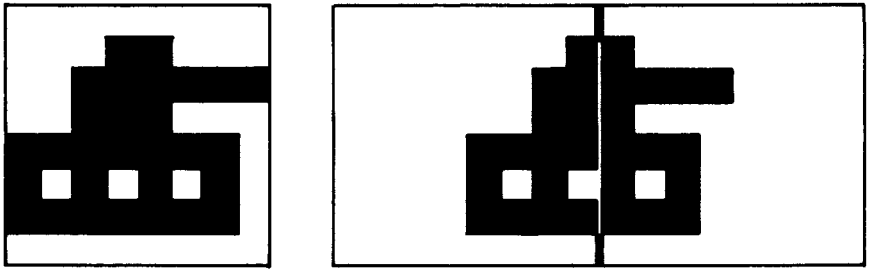
```
0 REM *** TYTTO ***:REM B4
10 PRINT "<CLR/HOME>":REM A3
20 PRINT "<HOME><5DOWN>   <RVS ON>F FK<RVS OFF>  ":REM
   C7
30 PRINT "    |. | |   ":REM DO
40 PRINT "    ‿ |   ":REM 3E
50 PRINT "   _|| /":REM 29
60 PRINT "  /  |":REM 41
70 PRINT "  ||  | ":REM 40
80 PRINT "  ||  | ":REM 41
90 PRINT "  /‾‾\ ":REM FA
100 PRINT "  |    | ":REM CO
110 PRINT "  |    | ":REM C1
120 PRINT "  ‾TT‾ ":REM 28
130 PRINT "   J L":REM EB
150 FOR VI=1 TO 100: NEXT VI:REM D6
200 PRINT "<HOME><5DOWN>   <RVS ON>F FK<RVS OFF>  ":RE
  M F7
210 PRINT "    |. |  / ":REM 1A
220 PRINT "    ‿ /  ":REM 88
230 PRINT "   _|| /":REM 59
240 PRINT "  /  <PUR>*<LT.BLUE>":REM DO
320 FOR VI=1 TO 100: NEXT VI:REM D5
330 GOTO 20:REM 31
```

4 Liikkuvat merkit

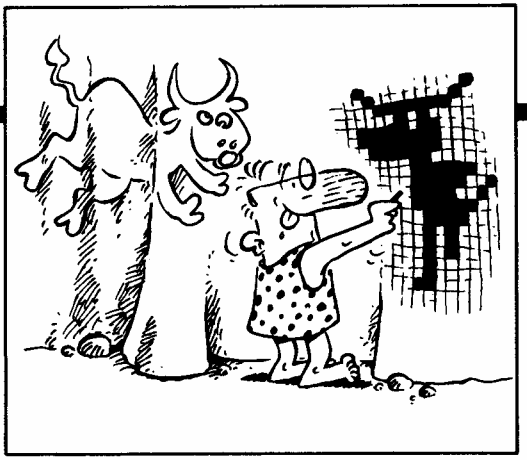
Merkeillä toteutettavassa animaatiossa on hyvä hyödyntää itse laadittuja merkkejä. Näin onkin tehty seuraavassa lisäyksessä edellisessä kappaleessa esitettyyn ohjelmaan, joka muutti -merkin avaruusraketiksi. Tämä lisäys saa raketin lentämään, mutta millä seurauksilla?

```
200 FORSI=0TO25: PRINT"<39SPACE>";: REM JUST I
  N CA:REM 51
210 NEXT SI:REM 6E
220 FOR SI=1994 TO 1034 STEP -40:REM 63
230 POKESI,0:POKESI+40,32:FORVI=1TO40:NEXTVI:R
  EM 2E
240 NEXT SI:REM 71
250 FOR SI=1 TO 10: FOR S=0 TO 7:REM 47
260 POKE S+12288,INT(RND(0)*256):REM DE
270 NEXT S,SI:REM F3
280 RESTORE: FOR SI=7 TO 0 STEP -1:REM 5C
290 READ ME: POKE SI+12288, ME: NEXT:REM B5
300 FOR SI=1034 TO 1994 STEP 40:REM 35
310 POKESI,0:POKESI-40,32:REM E2
320 FORVI=1TO(-1*((SI-1024)/40-26))↑1.5:NEXT V
  I,SI:REM 31
```

Omia merkkejä voidaan myös käyttää siten, että liikkuva hahmo muodostuu useasta vierekkäin tulostettavasta merkistä, jotka yhdessä muodostavat liikuteltavan hahmon.



Esimerkki merkkihahmojen liikkeen 'pehmentämisestä' käyttämällä puoliksi kahden merkin välille jaettuja merkkihahmoja.



5 Spritet

Kuusnelosen vahvin puoli pelikoneena ovat sen spritet. Niiden avulla on helppoa luoda liikkuvia hahmoja peleihin. Valitettavasti vain käytetty Basic-murre ei tue spritejen ohjelmointia, vaan kaikki on tehtävä pokeilemalla asianomaisiin rekistereihin.

Commodoren spritet muodostuvat 24×21 pisteestä. Niitä voidaan normaalisti pitää kerralla näytössä enintään kahdeksan. Spriteistä on määriteltävä muoto, koko ja väri sekä sen paikka näytöllä.

Spriten näytölläolo

Spritet numeroidaan nollasta seitsemään ja muistipaikkaan 53269 tallennetaan tieto siitä, mitkä spritet kulloinkin ovat näytöllä. Tämä on toteutettu siten, että kukin muistipaikan 53269 bitti vastaa aina yhtä spriteä. Jos bitti on asetettu ykköseksi, sitä vastaava sprite on näytöllä ja jos se on asetettu nollaksi, sprite ei ole näytöllä.

Kun näytölle sijoitetaan spritet 0, 3 ja 7 tulee muistipaikkaan 53269 tallentaa luku 2↑0 + 2↑3 + 2↑7 = 1+8+128 = 137. Kun lisätään tietty sprite näytölle, käytetään lausetta

POKE 53269, PEEK(53269) OR (2↑SP)

missä SP on näytölle tulevan spriten numero. Vastaavasti kun poiste-

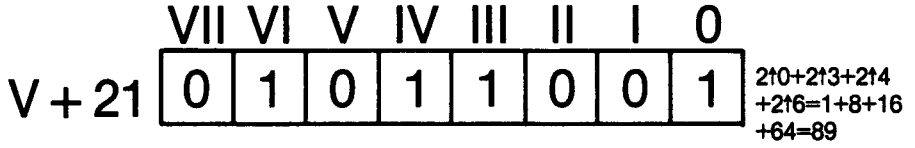
taan sprite näytöltä suoritetaan lause

POKE 53269, PEEK(53269) AND (255-2↑SP)

missä SP on näytöltä poistettavan spriten numero.
Esimerkiksi lisäystä ei siis saa tehdä seuraavasti:

POKE 53269, 2↑SP

koska tällöin kaikki muut, näytöllä ennestään olevat spritet katoavat.



Kun muistipaikan (rekisterin) V+21 sisältönä on 89, spritet 0, 3, 4 ja 6 ovat näytöllä.

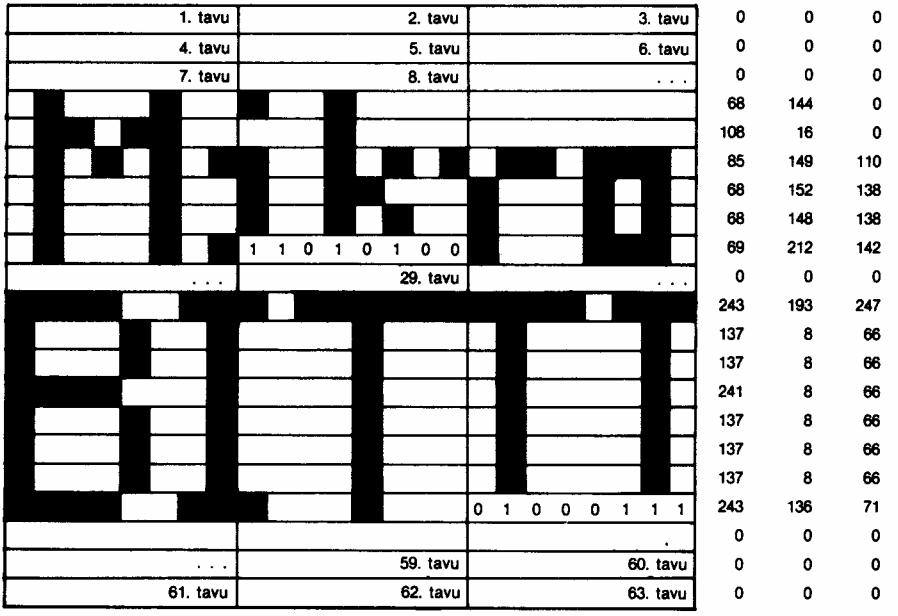
Spritejen yhteydessä käytetään usein apumuuttujaa V, jonka arvona on 53248. Tämä arvo osoittaa VIC-II -piirin sprite-rekisterien alkuosoitteeseen. Näin ei tarvitse muistaa pitkiä muistiosoitteita, vaan ainoastaan 53248:aan summattava arvo. Esimerkiksi edellä esitelty spritejen näytölläolon määräävä muistipaikka on tällä tavoin esitettynä V+21.

Sprite-lohko

Se, miltä sprite todellisuudessä näyttää riippuu spriten muodon määräävien 63 muistipaikan sisällöstä. Koska sprite koostuu 21 vaakarivissä olevista 24 pystysarakkeesta, voidaan helposti päätellä spriten muodostuvan 21:stä kolmen tavun rivistä. Tavu puolestaan sisältää kahdeksan bittiä, eli täyttää juuri yhden muistipaikan. Näin yksi rivi voidaan siis esittää kolmen muistipaikan sisällön avulla. Muistipaikan sisältöä vastaavassa binaariluvussa spriten pistettä vastaa ykkönen ja tyhjää, läpinäkyvää paikkaa nolla.

Tiedot on tallennettava muistiin rivi kerrallaan ylhäältä alas. Ensiksi tiedot ylärivin kahdeksasta ensimmäisestä pisteestä, sitten seuraavaan muistipaikkaan ylärivin kahdeksan keskimmäistä pistettä, sitten ylärivin

kahdeksan viimeistä pistettä, minkä jälkeen toisen rivin kahdeksan ensimmäistä pistettä jne. Kaikenkaikkiaan tarvitaan siis 63 muistipaikkaa, jotka muodostavat ns. sprite-lohkon.



Spriten suunnittelu ja miten sitä vastaavan muistilohkon sisältö lasketaan. Luvut talletetaan muistiin vaakariveittäin ylhäältä alas.

Se, mihin muistialueeseen spriten muodon määräävä sprite-lohko on sijoitettu ilmoitetaan muistipaikoissa 2040-2047, missä 2040 sisältää tiedon 0-spriten datojen sijainnista ja 2047 vastaavan tiedon 7-spriten osalta. Näissä muistipaikoissa ei kuitenkaan ole varsinaista muistialueen alun osoitetta. Koska niihin voidaan sijoittaa vain lukuja väliltä 0-255, niihin on sijoitettu sen muistilohkon numero, jossa sprite-lohko sijaitsee. Varsinainen alkuosoite saadaan kertomalla tämä muistilohkon numero 64:llä.

Sopivia muistilohkoja sprite-datojen talletukseen ovat 13, 14, 15 (kaset-

tipuskuri), 192, 193, ... Jos sprite-lohkon sijoittaa muistialueen pohjalle on pidettävä varansa, jottei pitkä Basic-ohjelma kirjoittuisi sprite-datojen päälle. Tarpeen vaatiessa muistia kannattaakin erikseen varata spritejä varten samaan tapaan kuin merkistöä muunnon yhteydessä

Spriten sijainti näytöllä

Spriten muodon lisäksi on tärkeätä tietää sen paikka näytöllä. Paikan määräämiseksi on tiedettävä spriten vaaka- ja pystykoordinaatit, jotka sijoitetaan asianomaisiin muistipaikkoihin.

Pystysuunnassa spriten sijainti näytöllä lasketaan ylhäältä alas. Valittavana on arvoja väliltä 0-250, jolloin arvon ollessa 0 tai 250 sprite on täysin näytön ulkopuolella — 'reunan takana piilossa'. Kun määritellään spriten paikka näytöllä pystysuunnassa se sijoitetaan muistipaikkaan V+1+2×SP, missä SP on spriten numero. Eli kun halutaan sijoittaa sijainniksi pystysuunnassa sata, se tehdään lauseella

POKE V+1+2×3, 100

Spritejen sijoittaminen vaakasuunnassa onkin jo hankalampaa, sillä valittavana on liki kolme ja puoli sataa eri paikkaa, eikä yhteen muistipaikkaan voi sijoittaa kuin 256:tta pienempiä lukuja. Näin ollen joudutaan menettelemään seuraavasti. Mikäli spriten osoite näytöllä vaakasuunnassa on pienempi kuin 256 sijoitetaan muistipaikkaan V+2×SP spriten paikka vaakasuunnassa ilman suurempia seremonioita. Jos sijainnin osoite kuitenkin kasvaa yli 255:n sijoitetaan muistipaikkaan V+2×SP luku SI-256 ja muistipaikkaan V+16, spriteä vastaavaksi bitiksi ykkönen ilmoittamaan osoiterekisterin ylivuodosta. 0-spriteä vastaa 0-bitti ja 7-spriteä 7-bitti.

Jos sijainti taas pienenee alle 256:n asetetaan V+16:ssa spriteä vastaava bitti nollaksi ja sijoitetaan muistipaikkaan V+2×SP spriten vaakasuuntaisen sijainnin todellinen arvo. Parhaiten tämä käy ilmi seuraavasta ohjelmasta, joka siirtää spriten näytön yli vasemmalta oikealle.

```
PROGRAM: ZEPPELIN

0 V=53248:REM C9
```

```
10 FOR SI=13*64TO14*64-2:REM 14
20 READ SD: POKE SI, SD:REM DD
30 NEXT SI: POKE V+21,PEEK(V+21)OR1:REM E3
40 POKE 2040,13: POKE V+1,100:REM C1
50 POKE  V+29,PEEK(V+29)OR1:REM E0
60 FOR SI=0 TO 344:REM 94
70 IF SI<256 THEN POKE V+16,PEEK(V+16)AND254:P
  OKEV,SI: GOTO90:REM 0C
80 POKEV+16,PEEK(V+16)OR1:POKEV,SI-256:REM 2C
90 POKE V+39, INT(RND(0)*16):REM E3
100 NEXT SI: POKEV+1,INT(RND(0)*179+50):REM AF
110 POKEV+27,INT(RND(0)*2): GOTO 60:REM AD
1000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,96,0,0,113:R
  EM 13
1010 DATA 255,224,119,255,248,124,209,20,253,9
  1,182,252,219,183,253,91:REM 81
1020 DATA 182,124,219,180,119,255,248,97,255,2
  24,0,31,128,0,31,0:REM 44
1030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:REM
  72
```

Spriteä muistipaikassa V+16 vastaava bitti saadaan samalla tavoin kuin spriten näytölläolon V+21:ssä määräävä bitti. Eli jos 5:n spriten x-osoite näytöllä kasvaa yli 255:n, tulisi V+16:sta sijoittaa

POKE V+16, PEEK(V+16) OR (2↑5)

Muita sprite-rekistereitä

Spritejen värit määrätään muistipaikkojen 53287-53294 avulla. Nämä värirekisterit sisältävät tiedon kunkin spriten väristä siten, että spriten 0 väri on muistipaikassa 53287 ja spriten 7 väri on muistipaikassa 53294. Apumuuttujan V avulla ilmaistuna tieto kunkin spriten väristä löytyy muistipaikasta V+39+SP, missä SP on kyseisen spriten numero (0-7). Värit määräytyvät muistipaikkojen sisällön (0-15) mukaan.

Kuusnelosen spritejä voidaan myös venyttää vaaka- ja pystysuunnassa.

Spriten koko voidaan kaksinkertaistaa joko jompaankumpaan tai molempiin suuntiin. Jos spritejä venytetään pystysuuntaan sijoitetaan muistipaikkaan V+23 venytettävien spritejen tiedot samaan tapaan kuin silloin kun kerrotaan tietokoneelle, mitkä spritet kulloinkin ovat näytössä. Eli jos halutaan venyttää sprite 2 pystysuunnassa, suoritetaan lause

POKE V+23, PEEK(V+23) OR (2↑2)

Vastaavasti jos spriten 6 venytys vaakasuunnassa lopetetaan, suoritetaan lause

POKE V+29, PEEK(V+29) AND (255-2↑6)

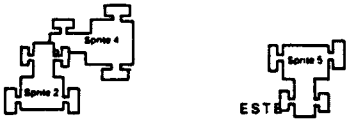
Eli tieto vaakasuuntaan venytettävistä spriteistä on muistipaikassa V+29.

Peliohjelmassa on hyvä tietää, onko näytöllä oleva sprite törmännyt toiseen spriteen tai taustalla olevaan tekstiin tai grafiikkapisteeseen. VIC-II-piirissä on onneksi rekisterit, joissa on tieto tällaisista tapahtumista.

Muistipaikka V+30 sisältää tiedon siitä, onko jonkin spriten näkyvä osa päällekkäin jonkin toisen spriten näkyvän osan kanssa. Eli jos spritet 2 ja 5 ovat päällekkäin, eivätkä muut spritet ole toistensa päällä, on muistipaikan V+30 sisältönä luku 2↑2 + 2↑5 = 36. Kun testataan, onko esimerkiksi sprite 3 päällekkäin jonkin toisen spriten kanssa, tapahtuu se seuraavasti

IF PEEK(V+30) AND (2↑3) = (2↑3) THEN ...

Muiden spritejen kanssa meneteltäessä toimitaan samoin, vain kakkosen potenssi vaihtuu. Potenssilasku voidaan kylläkin toki laskea valmiiksi, kuten muissakin tämän kirjan esimerkeissä.



	VII	VI	V	IV	III	II	I	0	
V+30	0	0	0	1	0	1	0	0	20
V+31	0	0	1	0	0	0	0	0	32

Kun näytöllä ei ole muita spritejä kuin 2, 4 ja 5 ja ne ovat toistensa ja tekstin päällä kuvan esittämällä tavalla, on törmäysrekisterissä taulukon esittämät arvot.

Ensinnäkin kaikki ohjelman tarvitsemat DATA-lauseet tulisi sijoittaa ohjelman, loppuun. Tällöin niiden läpikäyminen ei vie ohjelmalta kallista suoritusaikaa, kun ne eivät ole varsinaisen ohjelman seassa.

Samalla tavoin jos tutkitaan spritejen törmäyksiä taustaan tieto spriteistä, joiden näkyvät osat ovat taustan pisteiden päällä on talletettu muistipaikkaan V+31. Eli jos vain sprite numero 0 on taustan kanssa päällekkäin, on muistipaikan V+31 sisältönä 1 (=2↑0).

MUISTIPAIKKA		SEN MERKITYS
832—894		Muistilohko 13, kasettipuskuri, hyvä spritelohkon sijoituspaikka
896—958		Muistilohko 14, kasettipuskuri, hyvä spritelohkon sijoituspaikka
860—1022		Muistilohko 15, kasettipuskuri, hyvä spritelohkon sijoituspaikka
2040		0-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2041		1-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2042		2-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2043		3-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2044		4-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2045		5-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2046		6-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
2047		7-spriten datat sisältävän muistilohkon numero (13—15, 192—255)
53248	V	0-spriten x-osoite näytöllä
53249	V+1	0-spriten y-osoite näytöllä
53250	V+2	1-spriten x-osoite näytöllä
53251	V+3	1-spriten y-osoite näytöllä
53252	V+4	2-spriten x-osoite näytöllä
53253	V+5	2-spriten y-osoite näytöllä
53254	V+6	3-spriten x-osoite näytöllä
53255	V+7	3-spriten y-osoite näytöllä
53256	V+8	4-spriten x-osoite näytöllä
53257	V+9	4-spriten y-osoite näytöllä
53258	V+10	5-spriten x-osoite näytöllä
53259	V+11	5-spriten y-osoite näytöllä
53260	V+12	6-spriten x-osoite näytöllä
53261	V+13	6-spriten y-osoite näytöllä
53262	V+14	7-spriten x-osoite näytöllä
53263	V+15	7-spriten y-osoite näytöllä

MUISTIPAIKKA		SEN MERKITYS
53264	V+16	Spritejen x-osoitteiden ylivuotobitit (osoite >255). 0-spriten ylivuotobitti 0-bitissä jne.
53269	V+21	Tieto spritejen näytöllaolosta. Mikäli 5-sprite näkyvissä, niin 5-bitti on yksi.
53271	V+23	Spritejen venytys vaakasuunnassa. Kukin bitti merkitsee ollessaan ykkönen sitä, että vastaava sprite on venytetty.
53275	V+27	Mitkä spritet tulostuvat taustan päälle (kukin bitti vastaa yhtä spriteä).
53276	V+28	Tieto siitä, mitkä spritet ovat monivärisiä.
53277	V+29	Spritejen venytys pystysuunnassa.
53278	V+30	Ovatko spritet törmänneet toisiinsa (molempia (kaikkia) päällekkäin olevia spritejä vastaavat bitit ovat ykkösiä).
53279	V+31	Ovatko spritejen pisteet päällekkäin taustan pisteiden kanssa (kyseisiä spritejä vastaavat bitit ovat ykkösiä).
53285	V+37	Monivärispriterekisteri 1 (sisältää toisen monivärispriteille yhteisistä väreistä).
53286	V+38	Monivärispriterekisteri 2 (sisältää toisen monivärispriteille yhteisistä väreistä).
53287	V+39	0-spriten väri
53288	V+40	1-spriten väri
53289	V+41	2-spriten väri
53290	V+42	3-spriten väri
53291	V+43	4-spriten väri
53292	V+44	5-spriten väri
53293	V+45	6-spriten väri
53294	V+46	7-spriten väri
12288—12350		Muistilohko 192, hyvä sprite-lohkon sijoituspaikka. Voi joutua hyvin pitkän Basic-ohjelman alle.
12352—16383		Muistilohkot 193—255 hyviä sprite-lohkojen sijoituspaikkoja. Voivat joutua pitkän Basic-ohjelman alle.

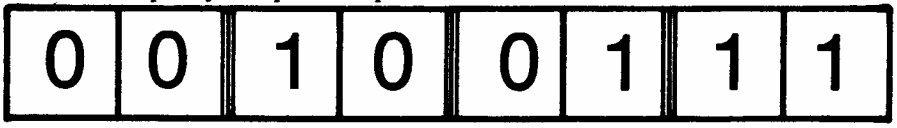
Moniväriset spritet

Moniväriset spritet ovat muilta osin samanlaisia kuin tavalliset spritet, paitsi että yhden pisteen päällä- tai poissaolon sijasta joudutaan kerralla määrittämään kahden pisteen tila yhdellä kertaa. Näiden kahden pisteen tilan määrittelemiseksi on kuitenkin käytettävissä kaksi bittiä. Kahden bitin avulla voidaan asianomaisen pisteparin pisteiden tila määritellä neljällä eri tavalla seuraavasti. Mikäli pisteparia vastaava bittipari on 00, on pistepari läpinäkyvä. Jos bittipari on 10, saadaan pisteparin väri tavalliseen tapaan spriten omasta värirekisteristä (V+39+SP). Jos bittipari on 01, saadaan pisteparin väri monivärispriterekisteristä V+37, ja vastaavasti jos bittipari on 11, saadaan pisteparin väri monivärispriterekisteristä V+38. Tämä on esitetty vielä seuraavassa taulukossa.

Bittipari	Vastaavan pisteparin tila
00	Läpinäkyvä, taustan väri
01	Väri monivärispriterekisteristä V + 37
10	Väri omasta värirekisteristä V + 39 + SP
11	Väri monivärispriterekisteristä V + 38

Yksi monivärispritejen väreistä voidaan siis määritellä spritekohtaisesti. Kaksi muuta käytettävää väriä ovat kuitenkin yhteisiä kaikkien monivärispritejen kesken.

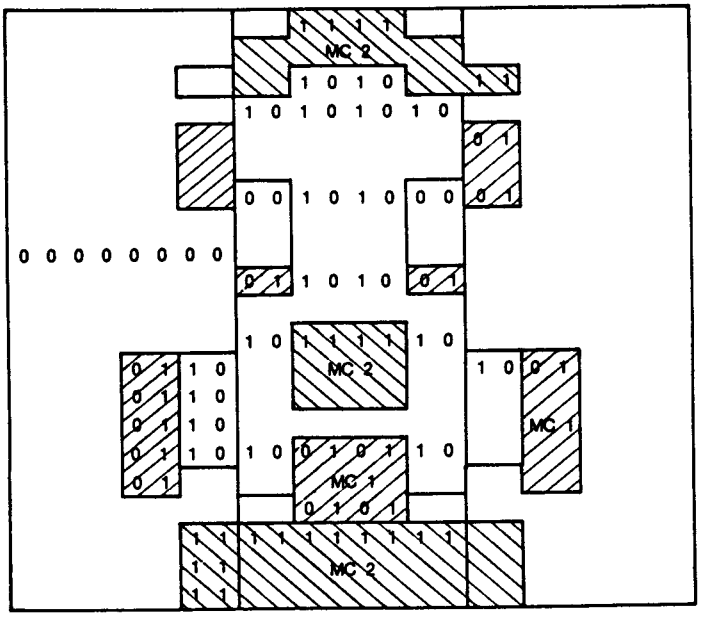
Monivärispritejen suurin haittapuoli on vähentynyt erottelytarkkuus. Vaakasuunnassa voidaan erotella vain 12 pistettä (pisteparia), joskin niille voidaan määrittää kolme eri väriä. Pystysuunnassa erottelutarkkuus on tavallisten spritejen tapaan 21 pistettä.



Tavallisessa sprite-lohkossa yllä-oleva tavu merkitsisi vain neljän pisteen päälläoloa ja neljän pisteen läpinäkyvyyttä. Moniväri-sen spriten kyseessä ollessa tavu kuitenkin jaettaisiin kahden bitin pareihin, joissa 00 merkitsisi vastaavan pisteparin näkyvyyttä, 10 pisteparin värin hakemista spriten omasta värirekisteristä, 01 pisteparin värin hakemista monivärispriterekisteristä V+37 ja 11 sitä, että pisteparin väri haetaan monivärispriterekisteristä V+38.

5 Spritet

Muistintarve on sama kuin normaaleilla spriteillä. Ensimmäisten neljän pisteparin väritiedot tallennetaan ensimmäiseen muistipaikkaan ja seuraavien neljän seuraavaan samaan tapaan kuin yksivärisissä spriteissä.

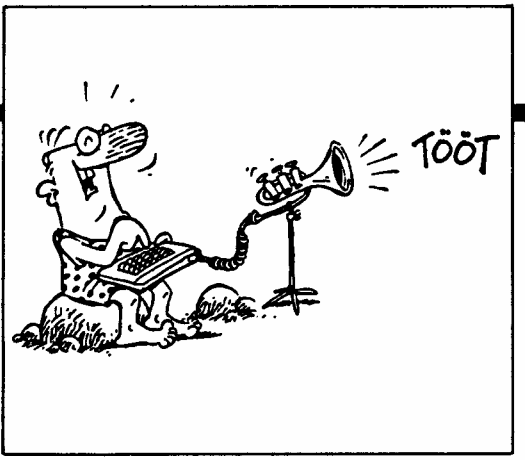


0	60	0
0	255	0
3	235	192
0	170	0
1	170	64
1	170	64
1	40	64
0	40	0
0	40	0
0	105	0
0	170	0
0	190	0
6	190	144
6	190	144
6	170	144
6	150	144
4	150	16
0	20	0
3	255	192
3	255	192
3	255	192

Monivärisen spriten suunnitteluesimerkki. Läpinäkyvä pistepari on 00, multicolor 1 01, multicolor 2 11 ja spriten oma väri 10.

Tieto siitä, mitkä näytöllä olevista spriteistä ovat monivärisiä on talletettu rekisteriin V+28 samaan tapaan kuin rekisteriin V+21 tieto siitä, onko jokin sprite käytössä vai ei. Eli kun halutaan spritejen 6 ja 7 olevan monivärisiä, sijoitetaan tieto rekisteriin seuraavanlaisesti

POKE V+28, (2↑6+2↑7)



6 Taustamusiikki ja äänitehosteet

SID äänipiiri tarjoaa seuraavat mahdollisuudet:

— 3 toisistaan riippumatonta äänigeneraattoria
— 4 aaltomuotoa: kolmioaalto, saha-aalto, neliöaalto ja valkoinen kohina.
— säädeltävissä oleva verhokäyrä ADSR
— kaikille äänille yhteinen monipuolinen suodatin
— kaikille äänille yhteinen voimakkuuden säätö
— prosessori voidaan kytkeä lukemaan 3:nnen äänen oskillaattoria ja verhokäyrägeneraattoria (modulointi)
— mahdollisuus syöttää piirille ulkopuolista audiosignaalia
— rengasmodulaattori
— synkronikytkentä

SID-piirin ohjaus:

Piirin toimintoja säätelevät muistipaikat 54272-54296. Koska C64:n BASICissa ei ole erikseen musiikkikäskyjä, on ohjeet annettava POKE-käskyillä.

Yleinen käskymuoto on: POKE xxxxx,yyy.

xxxxx on muistipaikan osoite (54272-54296), ja yyy on lukuarvo (0-255)

Suositeltava tapa on antaa jollekin muuttujalle arvoksi äänipiirin alkuosoite, ja osoittaa sen avulla muut muistipaikat Esim. jos S=54272, voidaan osoite 54275 merkitä lyhyesti S+3:lla.

Osa rekistereistä sisältää tietoa useammasta kuin yhdestä äänimuuttujasta ja toisaalta joidenkin muuttujien lukuarvojen esittäminen vaatii useampia bittejä kuin mitä yksi muistipaikka voi sisältää. Ensimmäisessä tapauksessa on järkevää laatia kaava, jonka avulla voidaan ohjata yksittäisiä bittejä. Toinen tapaus taas vaatii kaavan, joka jakaa halutun lukuarvon tarvittavien kahden tavun kesken (ns. HI-tavu ja LO-tavu).

Ennen äänitehtävän suorittamista on yleensä hyvä nollata piirin kaikki rekisterit (0-24), jolloin voi olla varma omien käskyjen toimivuudesta. Seuraava yhden rivin ohjelma hoitaa nollauksen:

```
10 S=54272:FORN=0TO24:POKES+N,0:NEXTN
```

Piirin nollaus on erityisesti tarpeen kohinaa käytettäessä, sillä tietyissä tilanteissa generaattorit saattavat lukkiutua.

Äänen korkeus

Äänen korkeuden ohjaamiseen käytetään kahta muistipaikkaa generaattoria kohden. HI-tavu huolehtii karkeasäädöstä ja LO-tavu hienosäädöstä. Joissakin tapauksissa on nopeuden vuoksi järkevää käyttää vain HI-tavua, esim. tehosteet, kun taas musiikki vaatii tarkempaa äänen korkeuden hallintaa.

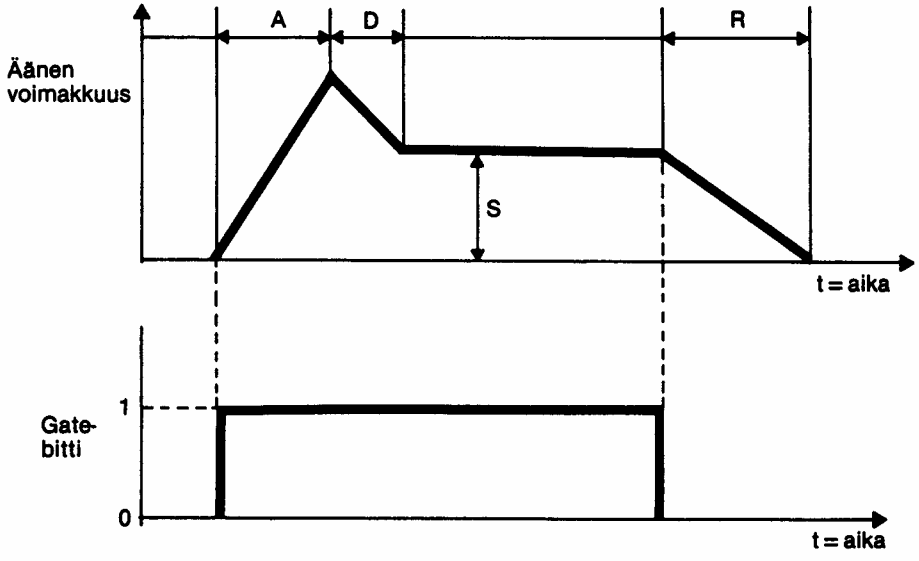
Nuottien sävelkorkeuksia vastaavia lukuarvoja on saatavissa käsikirjojen taulukoista: Joko muodossa kokonaisarvo (1-65535) tai eriteltynä taajuuden HI- ja LO-tavujen arvoiksi. Yleinen kaava halutun taajuuden lukuarvon NR saamiseksi on:

NR=F/0.0588235294, jossa F on taajuus. Taulukoissa esitetyt arvot antavat Suomessa oikeat sävelkorkeudet, kun NR arvot kerrotaan luvulla 1.03644174.

Numeroarvon ja HI- ja LO-tavujen välillä vallitsee seuraava yhteys:
HI=INT(NR/256) ja LO=NR-HI×256, NR=HI×256+LO

Verhokäyrä eli ADSR

ADSR kertoo äänen voimakkuuden vaihteluista yhden erillisen äänen aikana. Se koostuu seuraavista osista, joista kukin vaikuttaa ratkaisevalla tavalla äänen luonteeseen.



6 Äänitehosteet

A: Attack eli nousuaika (0-15)
— aika, jonka kuluessa äänen voimakkuus nousee suurimpaan arvoonsa
— lyömäsoittimilla, pianolla ja cembalolla on hyvin lyhyt nousuaika, samoin näppäiltävillä soittimilla kuten kitaralla, harpulla ja banjolla
— puupuhallinten kuten huilun ja klarinetin sekä jousisoitinten, kuten viulun ja sellon nousuaika on melko pitkä.
— vaskisoittimilla nousuaika on lyhyt
— laukauksten ja räjähdysten nousuaika on hyvin lyhyt

D: Decay eli tasaantumisaika (0-15)
— aika, jonka kuluessa äänen voimakkuus laskee huippuarvosta jälkisointitasolle.
— lyömäsoittimilla on yleensä lyhyehkö tasaantumisaika, jonka kuluessa äänen voimakkuus laskee nollaan
— banjon ääni hiljenee nopeasti ilman jälkikaiuntaa

S: Sustain eli jälkisointitaso (0-15)
— voimakkuus, jolla ääni soi niin kauan, kuin gate-bitti on päällä eli ykkönen
— lyömäsoittimilla ei yleensä ole lainkaan jälkisointitasoa.
— puhallinsoittimilla ja jousisoittimilla on korkea jälkisointivoimakkuus

R: Release eli laskuaika (0-15)
— aika, jonka kuluessa äänenvoimakkuus laskee jälkisointitasolta nollatasolle.
— jakso alkaa heti, kun gate-bitti saa arvon nolla. Tämä vastaa tilannetta, jossa soittaja lopettaa äänen. Sen jälkeen kuluu soittimesta riippuen jonkin aikaa ennenkuin ääni on kokonaan sammunut.
— pianon, cembalon ja urkujen ääni sammuu heti, kun kosketin nousee
— kitaran ja harpun kieli jää soimaan ja vaimenee hitaasti
— kaikuisassa tilassa ääni ei sammu heti, vaan vaimenee hitaasti

Jokaisen äänigeneraattorin ADSR on jaettu kahteen muistipaikkaan.
Tarvittavat ohjausarvot saadaan kaavoista:

AD=16×A+D ja **SR=16×S+R.**

Aaltomuodot

Erilaiset äänenvärit selittyvät sillä, että niillä on erilaiset ns. yläsävelsarjat. Äänen perustaajuuden kanssa yhtä aikaa aikaa soi joukko hyvin hiljaisia yläsäveliä, joita korva yleensä ei erota, vaan muodostaa niiden perusteella mielikuvan äänen väristä. Matemaattisesti voidaan tunnettujen yläsävelten perusteella laskea niiden yhteisvaikutuksena syntyvä aaltomuoto, ja on olemassa tiettyjä tyypillisiä aaltomuotoja, joita on helppo tuottaa elektronisesti. SID-piiri voi tuottaa seuraavanlaisia aaltomuotoja:

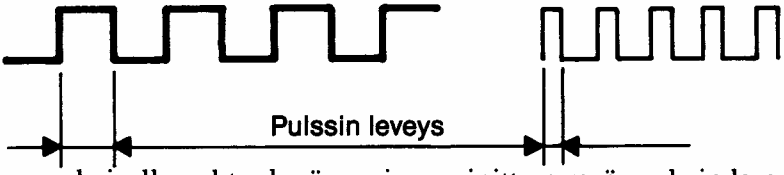
Kolmioaalto (triangle), kontrollirekisterin bitti 4 (16)

— vihellys, huilu ja muut puupuhaltimet

Saha-aalto (sawtooth) kontrollirekisterin bitti 5 (32)

— jousisoittimet

Pulssi-aalto (pulse) kontrollirekisterin bitti 6 (64)



— pulssiaallon yhteydessä on aina mainittava myös pulssin leveys. Se voi saada arvoja 0-4095. Usein pulssin leveys ilmoitetaan prosentteina, eli arvot vaihtelevat 0 ja 100 välillä. Tosin aivan ääriarvot eivät anna kunnollista ääntä.
— Haluttua pulssin leveysprosenttia P% vastaavat HI- ja LO-tavujen lukuarvot saadaan kaavoista:
HI=INT(0.159×P%) ja **LO=INT(0.159×P%)-HI**
— pulssin leveys vaikuttaa merkittävästi äänenväriin. Erilaisilla leveyksillä ja suodatuksella saa aikaan hyvinkin erilaisia sointeja:
— piano, cembalo, klarinetti, saksofoni, trumpetti ja säkkipilli
— pulssiaalto toimii hyvin yhdessä kolmioaallon kanssa

Kohina, valkoinen kohina (white noise) kontrollirekisterin bitti 7 (128)

— lyömäsoittimet
— suhina, kohina, humina, sihinä
— suihkukone, tuuli, meri
— läiskähdys,laukaus, jyräys Huom. Ei toimi hyvin muiden aaltomuotojen kanssa.
— myös kohinalle ilmoitetaan äänenkorkeuden HI ja LO arvot, vaikkakaan kohinalla ei ole mitään selvää äänenkorkeutta

Äänen voimakkuus

Äänen voimakkuutta V, säätelee muistipaikka 54296 (S+24), ja sen bitit 0-3.Voimakkuus voi saada arvot 0-15. Saman muistipaikan bitit 4-6 toimivat suodintyypin valintakytkiminä.

Suodattimet

Suodatin on systeemi, joka päästää valikoivasti läpi joukon ennalta määrättyjä taajuuksia tai taajuuskaistan, mutta poistaa muut. Suodattamalla yläsävelsarjaa, voidaan tehokkaasti muokata äänen väriä. Seuraavat tekijät ohjaavat suotimien toimintaa:

Rajataajuus (cut off frecuency): Taajuus, josta lähtien suodin alkaa leikata tai päästää taajuksia. Muistipaikat 54293 (S+21) LO (bitit 0-2), ja 54294 (S+22) HI (bitit 0-7) määräävät rajataajuuden RT, joka voi saada arvot 30-12000 Hz. HI- ja LO-tavujen lukuarvot saadaan kaavoista:

HI=INT((RT-30)/35.1) ja **LO=INT((RT-30)/35.1)-HI.**

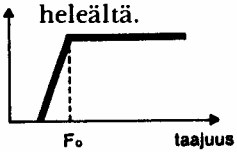
Resonanssi: Efekti, joka vahvistaa taajuuksia rajataajuuden tai keskitaajuuden lähellä saaden aikaan terävämmän äänen. Rekisterin 54295 (S+23) bitit 4-7 määräävät resonanssin voimakkuuden, joka voi saada arvot (0-15). Resonanssille RF saadaan kaava **POKES+23,RS×16.**

Resonanssin avulla voi saada aikaan lievän kaikuefektin.

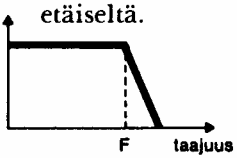
Suodatinkytkimet: Jokaiselle äänelle on oma bittikytkin, jonka avulla se on ohjattava kulkemaan suodattimen läpi. Nämä kytkimet sijaitsevat muistipaikan 54295 (S+23) biteissä 0-2

Suodatintyyppejä:

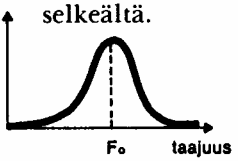
Ylipäästösuodin (high-pass), S+24, bitti 4, (16)
— leikkaa pois matalat taajuudet. Saa äänen kuulostamaan ohuelta ja heleältä.



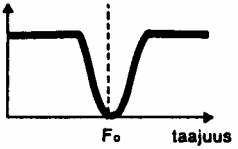
Alipäästösuodin (low-pass) S+24, bitti 5 (32)
— leikkaa pois korkeat taajuudet. Saa äänen kuulostamaan tummalta, ja etäiseltä.



Kaistanpäästösuodin (band-pass) S+24, bitti 6 (64)
— päästää läpi tietyn taajuuskaistan. Saa äänen kuulostamaan ohuelta ja selkeältä.



Kaistanestosuodin (notch) LP+HP, bitit 4 ja 6 (80)
— poistaa tietyn taajuuskaistan.
— muodostuu yhtäaikaa toimivista yli- ja alipäästösuotimista
Tällä suotimella voidaan poistaa esim. tietyllä taajuudella kuuluva hurinaa tai sirinä.



6 Äänitehosteet

Ehdotuksia eri soittimia muistuttaviksi ääniasetuksiksi:

AD=16×A+D ja SR=16×S+R

SOITIN	ADSR						AALTOM/	SUODATIN			OKT.
	A	D	AD	S	R	SR	PULSSI%	tyyp.	rajat	res.	
Piano	0	10	10	0	0	0	T	LP	5000	2	3-5
Cembalo	0	9	9	0	0	0	P10				3-5
Kitara	1	10	26	10	0	160	P20				4-5
Harppu	0	9	9	0	9	9	T				3
Banjo	0	6	6	0	0	0	P20	HP	5500	15	3
Bassokitara	0	10	10	0	9	9	P25	LP	5000	12	2
Huilu	6	6	102	6	6	102	T				4-5
Klarinetti	4	0	64	15	0	240	P50	LP	4500	4	3-4
Saksofoni	3	0	48	15	1	241	P30	LP	2000	0	2-3
Trumpetti	2	15	47	8	0	128	T	LP	1500	9	4-5
Pasuuna	2	15	47	8	0	128	T	LP	1500	15	2-3
Tuuba	3	15	63	8	4	132	T	LP	1500	8	1-2
Viulu	10	8	168	10	9	169	S	HP	12000	2	4-5
Sello	11	12	187	11	7	183	S	HP	5000	2	2-4
Urut	0	0	0	15	0	240	T				3-5
Hanuri	5	0	80	15	2	242	T	HP	12000	3	3-4
Säkkipilli	12	5	197	9	10	154	P10	LP	8000	0	4
Pikkurumpu	0	9	9	0	9	9	N	HP	1500	15	2
Kellopeli	0	7	7	0	11	11	T	HP	4000	13	6

Äänitehosteet

Yleisiä periaatteita
BASIC-kieliselle äänitehosteelle on ominaista se, että sen ajaksi kaikki muu toiminta keskeytyy. On kuitenkin mahdollista käyttää hyväksi tietoa, että äänipiiri toimii itsenäisesti, ja jatkaa aloitettua toimintaa ilman jatkuvaa ohjaustakin: Äänitapahtumalle annetaan parametrit eli siihen vaikuttavat muuttujat ja laukaistaan se käyntiin gate-bitillä. Nyt voidaan antaa äänen jatkua sellaisenaan tai äänipiirin sisäisen ohjauksen varassa, kunnes seuraava ohje ohjelmalta saadaan. Tätä menetelmää käyttäen saadaan tosin aikaan vain hyvin yksinkertaisia ääniefektejä.

Tehosteiden laatiminen edellyttää jo varsin pitkälle menevää tietoutta SID-piirin rekisterien ja bittien toiminnoista. Seuraava kuva esittelee kontrollirekisteriä, ja sen toimintoja.

KONTROLLIREKISTERI

0	0	0	1	0	0	0	1
KOHINA	PULSSI	SAHA	KOLMIO	TESTI BITTI	RING MOD BITTI	SYNC BITTI	GATE-BITTI
128	64	32	16	8	4	2	1
7	6	5	4	3	2	1	0

POKE 54275,17

Yllä olevasta kuvasta selviää kontrollirekisterin rakenne ja eri bittien vaikutus sen toimintaan. Jos halutaan soittaa kolmioaaltomuodolla jokin ääni, on asetettava ykkösiksi aaltomuotobitti 4 ja gate-bitti eli bitti 0. Näistä saadaan rekisterin binäärikoodiksi 00010001, joka on desimaaliluvuksi muutettuna: 16+1=17. Siis POKE 54276,17 tuottaa halutun tuloksen.

Kun alat suunnitella tehostetta, mieti mitä se kuvaa. Liittyykö ääni suoraan johonkin kuvaruudulla tapahtuvaan liikkeeseen, onko se moottorin käyntiääni tai laukaus, tai ääni joka syntyy yhteentörmäyksessä. Seuraavaksi selvitä itsellesi saako tehoste pysäyttää tapahtumat kuvaruudulla hetkeksi, vai olisiko syytä pyrkiä häiriöttömään keinoon. Esimerkkitehosteista 1,2,ja 6 toimivat BASIC-ohjelman kulkua haittaamatta.

6 Äänitehosteet

Riittää kun SID-piiri saa alkutiedot ja gate-bitti asetetaan ykköseksi. Monimutkaisemmat äänitapahtumat vaativat jatkuvaa ohjausta, jolloin muu tapahtuminen pysähtyy siksi aikaa. Tämäkin saattaa olla paikallaan esimerkiksi voittofanfaarina uuden ennätyksen kunniaksi tai ohjelman alkutunnuksena.

Seuraavissa tehosteissa on havainnollisuuden vuoksi käytetty POKE-lauseissa numeroiden yhteenlaskua. Varsinaisessa tehostekäytössä on suositeltavaa käyttää niin tiivistä esitystapaa kuin mahdollista ja jättää pois kaikki tyhjät välilyönnit ja laskutoimitukset.

```
5 REM RÄJÄHDYS
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+5,2X16+13:POKES+24,15
30 POKES+1,4:POKES,119:POKES+4,128+1
```

S+5 asettaa A:n ja D:n, S+24 määrää voimakkuudeksi 15, S+1 ja S määräävät kohinan korkeuden ja S+4 kytkee kohinan päälle.

```
5 REM KILAHDUS
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+24,15:POKES+1,110:POKES+5,9:POKES+6,9:POKES+4,16
30 FORT=1TO1000:NEXT:GOTO20
```

Kilahdus käyttää kolmioaaltoa:S+4=16, verhokäyrän nousuaika on nopea (A=0) , tasaantuminen alkaa hitaasti (D=9): S+5=9, jälkisointitasoa ei ole (S=0) ja laskuaika on pitkä (R=9): S+6=9. S+4,17 kytkee äänen päälle, ja heti sen jälkeen S+4,16 kytkee gate-bitin nollaksi, jolloin laskuaika alkaa. Rivillä 20 oleva aikaviive määrää kilahdusten toistoajan.

```
5 REM KIERRENOUSU/KAARINOUSU
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+5,15:POKES+6,9:POKES+24,15
30 POKES+15,13:POKES+4,16+4+1
40 FORF=0TO255:POKES+1,F:POKES,F:NEXT
50 POKES+4,20
```

Tässä tehosteessa käytetään rengasmodulaattoria, jolloin ääni 1 on kytketty ääni 3:n kanssa syöttämään lähtötaajuuksia modulaattorille.

S+5 ja S+6 määräävät äänen 1 verhokäyrän, S+15 määrää äänen 3 taajuuden HI-arvon. S+4,16+4+1 määrää aaltomuodoksi kolmioaallon (16), kytkee RING MOD- eli rengasmodulaattori-bitin (4) ja gate-bitin (1) ykkösiksi. Rivillä 40 ääni 1:n taajuusarvot kasvavat nollasta maksimiin: S+1 ja S asettavat HI- ja LO-arvoiksi F:n.

Rivillä 50 S+4 kytkee gate-bitin nollaksi aaltomuodon ja RING-bitin jäädessä ennalleen.

Jos asetat rivillä 30 S+4 aaltomuodoksi saha-aallon (32), muuttuu äänen luonne kieppuvasta suoraksi.

```
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+5,3X16+9:POKES+24,15
30 FORN=1TO30:POKES+4,16+1
40 FORF=INT(RND(0)X90)+10 TO INT(RND(0)X150)+60
   STEP2:POKES+1,F:NEXT
50 POKES+4,16:NEXTN
```

S+5 asettaa A:n ja D:n, S+24 määrää äänen voimakkuudeksi 15. Rivillä 30 asetetaan silmukka, joka toistaa tapahtuman 30 kertaa, S+4 asettaa aaltomuodon ja kytkee gate-bitin ykköstilaan. Rivillä 40 arvotaan RND-funktion avulla kaksi taajuusarvoa, ja silmukassa liu'utaan alarajalta ylärajalle käyttäen vain taajuuden HI-arvoa..

```
 5 REM PAPUKAIJA
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+5,10:POKES+6,16X5=5
30 POKES+24,15:POKES+4,16+1:FORN=1TO20
40 FORF=50TO200STEP12:POKES+1,F:NEXT
50 NEXT N
60 FORN=1TO15
70 FORF=50TO200STEP20:POKES+1,F:NEXT
80 NEXT N
90 FORF=60TO170:POKES+1,F:NEXT
100 FORF=70TO255:POKES+1,F:NEXT:POKES+4,16
```

Papukaija on toteutettu niin, että soitetaan ensin 20 kertaa nopeasti ja sitten 15 kertaa vähän hitaammin riveillä 40 ja 70 silmukoissa olevat taa-

juudet äänen 1 HI-arvoa käyttäen. Nopeuteen vaikuttaa STEP-komennon askelen suuruus. Lopuksi vielä kaksi silmukoilla toteutettua eri korkeuksilta alkavaa ylöspäin liukuvaa ääntä.

```
 5 REM MOOTTORI
10 S=54272:FORL=0TO24:POKES+L,0:NEXT
20 POKES+24,15:POKES+6,16×15:POKES+13,16×15
30 POKES+1,3:POKES+8,2:POKES+7,159
40 POKES+4,33:POKES+11,33
```

S+6 ja S+13 asettavat äänien 1 ja 2 SUSTAIN/RELEASE rekistereihin SUSTAIN-arvoksi 15. Rivillä 30 asetetaan yhtä aikaa soimaan kaksi matalaa ääntä, jolloin niiden yhteisvaikutuksesta syntyy muriseva moottorin ääni. Kokeile S+7 rekisteriin arvoja 50,100 ja 150.

SID rekisterikartta

OSOITE	REKISTERI
54272 S	ääni 1, taajuus / LO (0-255)
54273 S+1	ääni 1, taajuus / HI (0-255)
54274 S+2	ääni 1, pulssin leveys / LO (0-255)
54275 S+3	ääni 1, pulssin leveys / HI (0-15)
54276 S+4	ääni 1, kontrolli & aaltomuodot:

bitti 0:	GATE on/off
bitti 1:	SYNC on/off
bitti 2:	RING MOD on/off
bitti 3:	TEST on/off
bitti 4:	kolmioaalto (16+1)
bitti 5:	saha-aalto (32+1)
bitti 6:	pulssiaalto (64+1)
bitti 7:	kohina (128+1)

54277 S+5	ääni 1, ATTACK/DECAY
54278 S+6	ääni 1, SUSTAIN/RELEASE
54279 S+7	ääni 2, taajuus / LO

54280 S+8 ääni 2, taajuus / HI
54281 S+9 ääni 2, pulssin leveys / LO, bitit 0-3
54282 S+10 ääni 2, pulssin leveys / HI
54283 S+11 ääni 2, kontrolli & aaltomuoto (Kts. ääni 1)
54284 S+12 ääni 2, ATTACK/DECAY
54285 S+13 ääni 2, SUSTAIN/RELEASE

54286 S+14 ääni 3, taajuus / LO
54287 S+15 ääni 3, taajuus / HI
54288 S+16 ääni 3, pulssin leveys / LO
54289 S+17 ääni 3, pulssin leveys / HI
54290 S+18 ääni 3, kontrolli & aaltomuoto (kts. ääni 1)

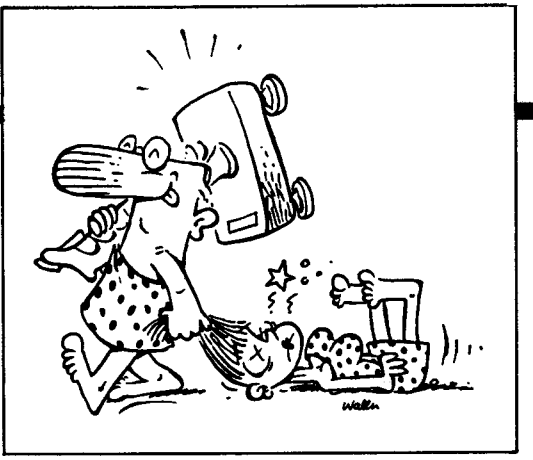
54291 S+19 ääni 3, ATTAC/DECAY
54292 S+20 ääni 3, SUSTAIN/RELEASE
54293 S+21 suodattimen rajataajuus / LO, bitit 0-2
54294 S+22 suodattimen rajataajuus / HI, bitit 0-7
54295 S+23 suodattimen asetukset:

bitti 0: suodatin on/off ääni 1
bitti 1: suodatin on/off ääni 2
bitti 2: suodatin on/off ääni 3
bitti 3: suodatin on/off ulkopuolinen signaali
bitit 4-7: resonanssi (0-15)

54296 S+24 äänien yhteinen voimakkuus / suodatuslajit

bitit 0-3: äänien yhteinen voimakkuus
bitti 4: alipäästösuodin (Low-Pass)
bitti 5: kaistanpäästösuodin (Band-Pass)
bitti 6: ylipäästösuodin (High-Pass)
bitti 7: ääni 3:n poiskytkentä esim. modulointikäytössä.

7



Ohjaus joystickillä

Joystickeissä on viisi kytkintä, joista neljä osoittaa suuntaa ja yksi toimii normaalisti tulistusnäppäimenä. Välisuunnat saadaan kahden suuntaa osoittavan kytkimen yhtäaikaisesta kytkeytymisestä.

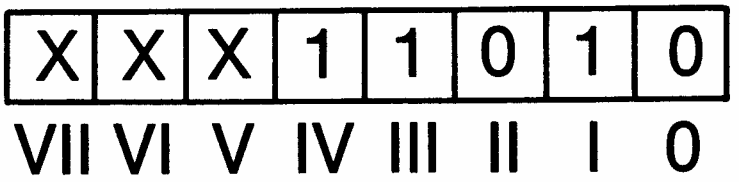
Tietokoneen sisällä A-porttiin sijoitetun joystickin tila on luettavissa muistipaikassa 56320 olevasta rekisteristä. Vastaavasti B-porttia vastaa muistipaikka 56321.

Muistipaikassaan jokainen joystickin kytkimistä vastaa yhtä bittiä. Ne on järjestetty seuraavasti: Ylös — nollabitti, alas — ykkösbitti, vasemmalle — kakkosbitti, oikealle — kolmosbitti ja tulta — nelosbitti. Kytkimiä vastaavat bitit miehittävät siis muistipaikkansa seuraavan taulukon osoittamaan tapaan.

X	X	X	✸	→	←	↓	↑
VII	VI	V	IV	III	II	I	0

56320 tai
56321

Jos halutaan tietää, onko joystickiä väännetty johonkin suuntaan, on muistettava, että mikäli kyseistä kytkintä ei ole painettu, kytkintä muistipaikassa vastaava bitti on ykkönen. Jos kytkin on kytketty on sitä vastaava bitti muistissa siis nolla.



Tilanne, kun A-porttiin kytkettyä joystickiä väännetään ylös vasemmalle.

Käytännössä kun tutkitaan onko B-porttiin kytketyn joystickin firenäppäintä painettu, voidaan se testata seuraavankaltaisella lauseella

IF PEEK(56321) AND (2↑4) = 0 THEN ...

Kätevämmin B-porttiin kytketyn joystickin tilan tutkiminen kuitenkin käy päinsä seuraavaan tapaan

```
10 JV=PEEK(56320):FR=JVAND16:JV=15-(JVAND15)
```

minkä jälkeen muuttujalla JS on väännöstä riippuen seuraavanlaiset arvot:

JS	Suunta
1	Ylös
2	Alas
4	Vasemmalle
5	Ylös vasemmalle
6	Alas vasemmalle
8	Oikealle
9	Ylös oikealle
10	Alas oikealle

Muuttuja FR on 0, mikäli tulitusnäppäintä on painettu, ja 16, jos sitä ei ole painettu.

Kun tutkitaan A-porttiin kytkettyä joystickiä käytetään PEEK-lauseessa osoitetta 56320.

Vielä helpommin joystickin luku käy seuraavan ohjelman avulla. Mi-

7 Ohjaus joystickillä

käli ohjelmassa aiotaan käyttää joystickiä, kannattaa tämä ohjelmanpätkä sijoittaa aivan ohjelman alkuun (tai loppuun, mihin ohjelman alusta sitten hypätään). Tällöin ohjelmassa myöhemmin joystickin suuntaa tarvittaessa ne on sijoitettu muuttujaan STICK seuraavan taulukon mukaisesti.

STICK	Suunta
—40	Ylös
40	Alas
— 1	Vasemmalle
1	Oikealle

Fireä painettaessa muistipaikka 828
on yksi, muutoin se on nolla.

```
9900 REM JOYSTICKIN LUKU (CREDITS TO PANU HAAR
  AMO):REM 2D
9910 FOR T=49152TO49245:READA:POKET,A:NEXT:REM
   35
9920 SYS49233:REM D8
9930 DATA 24,173,,220,41,1,240,23,173,,220,41,
  8,240,22,173,,220,41,2,240,21,173:REM FD
9940 DATA ,220,41,4,240,20,144,44,169,216,133,
  144,144,16,169,1,133:REM A7
9950 DATA 144,144,10,169:REM A8
9960 DATA 40,133,144,144,4,169,255,133,144,173
  ,,220,41,16,240,8,169,,141:REM D6
9970 DATA 60,3,76,49:REM EA
9980 DATA 234,169,1,141,60,3,144,246,169,,133,
  144,144,228,120,169,,141:REM 7C
9990 DATA 20,3,169,192,141,21,3,88,96:REM 31
```

8

Peliohjelmat

Lunar Adventure

Teemu Stenberg

Lunar Adventure on toimintapeli, jossa pelaajan tehtävänä on ohjata avaruusaluksensa monimutkaisen kuussa sijaitsevan luolakompleksin läpi. Luolissa on erilaisia ansoja, ja laskeuduttaessa on oltava varovainen, jotta laskeutumisnopeus ei kasvaisi liian suureksi.
Alusta ohjataan seuraavista näppäimistä:

— F1 syöttää polttoainetta moottoriin, jolloin nousunopeus kasvaa tai putoamisnopeus hidastuu.
— A sytyttää vasemmanpuoleisen ohjausraketin, jolloin alus alkaa liikkua oikealle.
— D sytyttää oikeanpuoleisen ohjausraketin, jolloin alus alkaa liikkua vasemmalle.

Ohjelma on jaettu kahteen osaan, jotka molemmat on ajettava, jotta ohjelma toimisi kunnolla.

Ensimmäinen ohjelma muuttaa Kuusnelosen merkistön ohjelman tarpeiden mukaan. Näin saadaan luotua 'futuristinen' merkistö ja taustat sekä jotkin tehosteetkin.

Toinen listaus pitää puolestaan sisällään varsinaisen ohjelman, joka hyödyntää ensiksi ladatussa ohjelmassa määriteltyä uutta merkistöä.

Kuva ohjelman näytöstä on kirjan kannessa vasemmalla puolella.

Ohjelmat kannattaakin sijoittaa samalle levykkeelle tai nauhalle peräkkäin. Ensiksi siis ladataan ja ajetaan ensimmäisessä listauksessa esitelty ohjelma, minkä jälkeen vasta varsinainen peliohjelma.

Ensimmäisen ohjelman ajamisen jälkeen ei saa painaa RUN/STOP ja RESTORE -näppäimiä yht'aikaa, jottei Kuusnelonen siirtyisi käyttämään normaalimerkistöä itse määriteltyjen merkkien sijaan.

```
PROGRAM:LUNAR ADVENTURE
5 PRINT"<WHT><CLR/HOME><5DOWN><LT.RED><13RIGHT>PLE
  ASE WAIT":POKE53280,0:POKE53281,0:REM 80
6 PRINT"<DOWN><5RIGHT>LUNAR ADVENTURE BY TEEMU STE
  NBERG.":REM 7C
7 PRINT"<DOWN><15RIGHT>21.10.1984":REM C2
20 POKE56334,PEEK(56334)AND254:REM AB
30 POKE1,PEEK(1)AND251:REM 01
40 FORI=0TO2040:POKEI+28672,PEEK(I+53248):NEXT:REM
   E5
50 POKE1,PEEK(1)OR4:REM 6D
60 POKE56334,PEEK(56334)OR1:REM 13
65 REM  **  KIRJAIMET  **:REM 97
70 FORZ=28672TO28672+215:READX:POKEZ,X:NEXT:REM CB
80 DATA0,0,0,0,1,0,0,0:REM 37
85 DATA126,102,102,102,254,230,230,230:REM 67
90 DATA124,108,108,108,254,230,230,254:REM 79
95 DATA126,98,96,96,224,224,226,254:REM 09
100 DATA120,108,102,102,230,230,230,254:REM 8B
105 DATA126,96,96,96,254,224,224,254:REM 30
110 DATA126,96,96,96,248,224,224,224:REM 2C
115 DATA126,98,96,96,230,226,226,254:REM 31
120 DATA102,102,102,102,254,230,230,230:REM 87
125 DATA24,24,24,24,56,56,56,56:REM 2A
130 DATA12,12,12,12,14,14,78,126:REM 40
135 DATA100,100,100,104,252,226,226,226:REM 96
140 DATA96,96,96,96,224,224,224,254:REM 02
145 DATA119,107,107,99,227,227,227,227:REM 91
150 DATA126,102,102,102,230,230,230,230:REM 8A
```

```
155 DATA126,102,102,102,230,230,230,254:REM 95
160 DATA126,102,102,126,224,224,224,224:REM 9D
165 DATA126,102,102,102,118,118,118,126:REM A3
170 DATA126,102,102,126,240,232,228,226:REM A1
175 DATA126,98,96,96,126,6,198,254:REM DF
180 DATA126,24,24,24,56,56,56,56:REM 5E
185 DATA102,102,102,102,230,230,230,254:REM 92
190 DATA102,102,102,102,230,230,236,248:REM 97
195 DATA219,219,219,219,251,251,251,255:REM C1
200 DATA102,102,102,24,24,230,230,230:REM 24
205 DATA102,102,36,24,56,56,56,56:REM 86
210 DATA127,99,6,12,24,48,99,127:REM 64
220 REM ** NUMEROT  **:REM 4A
230 FORZ=28672+384TO28672+463:READX:POKEZ,X:NEXT:R
  EM C8
235 DATA126,110,110,118,246,230,230,254:REM A0
240 DATA24,24,24,24,28,28,28,28:REM 24
245 DATA126,70,6,6,126,112,114,126:REM B1
250 DATA126,6,6,6,126,14,14,126:REM 1C
255 DATA102,102,102,126,6,6,6,6:REM 14
260 DATA126,98,96,96,126,14,78,126:REM D4
265 DATA126,98,96,96,254,230,230,254:REM 33
270 DATA126,70,6,6,14,14,14,14:REM E7
275 DATA126,102,102,102,254,230,230,254:REM 9E
280 DATA126,102,102,126,14,14,78,126:REM 12
285 REM  ** MAAASTO  **:REM 31
290 FORZ=28672+512TO28672+663:READX:POKEZ,X:NEXT:R
  EM C9
295 DATA192,224,224,240,252,252,254,255:REM B9
300 DATA3,7,7,15,63,63,127,255:REM F0
305 DATA192,192,224,240,240,224,248,248:REM B6
310 DATA248,248,240,248,252,252,254,255:REM BB
315 DATA1,3,7,7,15,15,31,31:REM 4D
320 DATA15,63,63,31,63,127,127,255:REM B8
325 DATA129,207,255,255,255,255,255,255:REM C5
330 DATA128,201,255,255,255,255,255,255:REM BA
335 DATA255,254,252,252,254,255,255,254:REM C0
340 DATA254,254,252,254,255,254,254,255:REM BD
345 DATA255,127,63,127,255,127,63,127:REM 5C
350 DATA255,127,127,63,63,127,63,127:REM 25
```

```
355 DATA255,255,255,255,255,255,239,108:REM CA
360 DATA255,255,255,255,255,255,179,17:REM 98
365 DATA255,255,255,255,255,255,255,255:REM CC
370 DATA0,0,0,0,7,31,127,255:REM 79
375 DATA1,7,31,127,255,255,255,255:REM C3
380 DATA128,224,240,254,255,255,255,255:REM BD
385 DATA0,0,0,0,192,240,254,255:REM 17
390 REM  ** SPRAITIT  **:REM 98
395 FORZ=31744TO31744+(64*7):READX:POKEZ,X:NEXT
   :REM 7D
398 REM  **  MODULI  **:REM FA
400 DATA1,255,128,1,255,128,9,129,144,31,129,248,1
  1,195,208,2,66,64,7,255,224:REM 08
405 DATA7,60,224,15,60,240,11,24,208,11,102,208,17
  ,195,200,17,255,200,22,126,104:REM 6C
410 DATA56,255,28,33,255,132,96,0,6,240,0,15,96,0,
  6,0,0,0,0,0,0,0:REM 7A
411 REM  **  SUIHKU  **:REM FB
415 DATA0,215,0,0,234,0,0,126,0,0,124,0,0,60,0,0,2
  4,0,0,8,0,0,8,0:REM 45
420 DATA,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,:R
  EM 90
425 REM  **  LIITAJA  **:REM 25
430 DATA,,,,,,,,,,,,,,,,,0,60,0,0,255,0,7,255,224,
  7,90,224,6,189,96,13,90,176:REM A0
435 DATA26,189,88,53,126,172,63,255,252:REM C9
440 DATA63,255,252,48,153,12,48,189,12,24,255,24,6
  ,129,96,1,129,128,0:REM 82
445 REM  **  RAJAHDYKSET  **:REM 63
446 REM   1:REM B3
450 DATA,,,,,,,,,,,,,,,,,,,,,,,,0,56,0,0,126,0,0,1
  26,0,0,255,0,0,127,0,0,254,0:REM 6D
451 DATA0,252,0,0,28,0,,,,,,,,,,,,,,,:REM 13
452 REM   2:REM B1
455 DATA,,,,,,,,,,,,,,,,,,0,0,128,1,199,192,0,239,
  224,0,255,224,1,255,192:REM D6
456 DATA1,255,224,3,255,224,1,255,224,1,127,240,0,
  127,240,0,126,96,0,17,0,,,:REM 9F
457 DATA,,,,,,:REM C2
460 REM   3:REM B1
```

```
461 DATA,,,,,,,,,,,,1,195,0,3,239,192,7,255,224,7,
  255,224,31,255,224,63,255,192:REM 2D
462 DATA63,255,224,31,255,224,15,255,248,3,255,248
  ,3,255,240,3,255,240,7,255,192:REM A1
463 DATA1,255,0,1,204,0,,,,,,,:REM BB
465 REM    4:REM B7
466 DATA0,6,0,0,255,0,0,255,24,0,127,248,30,127,25
  2,63,255,254,63,255,254:REM 2C
467 DATA31,255,255,31,255,255,127,255,254,127,255,
  252,127,255,248,127,255,252:REM 22
468 DATA63,255,252,63,255,252,31,255,248,31,255,22
  4,31,255,224,31,255,224:REM 4A
469 DATA15,255,224,0,241,128,0,0:REM 4D
480 FORT=31744+(64*7)TO31744+(64*8):POKET,255:NEXT
  :REM 4A
485 FORT=31744+(64*8)TO31744+(64*9):POKET,0:NEXT:P
  OKE32287,255:REM 1C
490 FORT=31744+(64*9)TO31744+(64*10)STEP2:POKET,0:
  POKET,255:NEXT:REM FD
492 FORT=31744+(64*10)TO31744+(64*11):POKET,255:NE
  XT:REM A1
495 FORT=28672+664TO28672+711:POKET,255-PEEK(T-152
  ):NEXT:REM 18
496 FORT=16000TO16010:READY:POKET,Y:NEXT:FORT=1602
  0TO16030:READY:POKET,Y:NEXT:REM C9
497 FORT=16040TO16050:READY:POKET,Y:NEXT:REM 20
498 DATA169,7,141,21,208,169,2,141,1,212,96,169,5,
  141,21,208,169,20,141,1,212,96:REM 97
499 DATA169,5,141,21,208,169,0,141,1,212,96:REM 7B
500 POKE53272,(PEEK(53272)AND240)+12:REM B4
501 POKE56578,PEEK(56578)OR3:REM 59
502 POKE56576,(PEEK(56576)AND252)OR2:REM 12
503 POKE53272,(PEEK(53272)AND15)OR176:REM 38
504 POKE648,108:PRINT"<CLR/HOME><GRN>":REM EB
505 FORT=26624TO27624:POKET,32:NEXT:REM 85
```

PROGRAM:LUNA

```
1 POKE53272,(PEEK(53272)AND240)+12:POKE53269,0:CLR
  :GOTO50:REM 11
2 POKE56578,PEEK(56578)OR3:REM F5
3 POKE56576,(PEEK(56576)AND252)OR2:REM AE
4 POKE53272,(PEEK(53272)AND15)OR176:REM D4
5 POKE648,108:REM CB
6 PRINT"<WHT><CLR/HOME><WHT>":POKE53280,0:POKE5328
  1,0:REM 33
7 POKE53272,(PEEK(53272)AND15)OR160:REM DO
8 FORT=1TO150:POKE27648+INT(RND(1)*1000),0:NEXT:SH
  =3:GR=.05:PRINT"<UPPER/LOWER OFF>":REM 16
9 PRINT"<6DOWN><8RIGHT><GRN>LUNAR ADVENTURE":REM A
  6
10 PRINT"<4DOWN><9RIGHT><YEL>PRESS F1 TO START,AL"
  :REM 07
11 POKE53272,(PEEK(53272)AND15)OR176:REM 02
12 V=53248:POKEV+21,3:POKEV,40:POKEV+1,140:POKEV+3
  9,12:POKEV+40,1:POKE28664,240:REM 98
13 POKEV+2,40:POKEV+3,156:POKE28665,241:REM F2
14 PRINT"<PUR><3DOWN><6RIGHT> <RVS ON>A<RVS OFF>-R
  IGHT <6RIGHT> <RVS ON>D<RVS OFF>-LEFT <25RIGHT>
  <RVS ON>F1<RVS OFF>-UP":REM F3
16 POKE28676,1:FORT=1TO4:NEXT:IFPEEK(197)=4THENGOT
  0100:REM 7D
17 POKE28676,3:FORT=1TO4:NEXT:IFPEEK(197)=4THENGOT
  0100:REM 80
18 GOTO16:REM 09
50 DIMQ(30):Q(1)=1:Q(2)=5:Q(3)=37:Q(4)=103:Q(5)=11
  9:Q(6)=247:Q(7)=255:REM 97
55 Q(8)=255:Q(9)=247:Q(10)=119:Q(11)=103:Q(12)=37:
  Q(13)=5:Q(14)=1:REM 77
60 FORT=15TO30:Q(T)=0:NEXTT:REM B4
70 GOTO2:REM D2
100 POKEV,40:POKEV+1,70:POKEV+2,0:POKEV+3,0:POKEV+
  21,0:LU=0:REM 36
101 PRINT"<CLR/HOME><12DOWN><14RIGHT><BLK>PLAYER 1
  ":POKE54276,0:POKE54273,0:REM 7D
```

```
102 POKE54296,30:POKE54295,1:POKE54294,80:REM 2F
105 POKE54277,0:POKE54278,255:POKE54276,129:RESTOR
  E:REM FC
110 FORI=0TO15:READY:FORG=0TO5:NEXTG:REM 54
111 FORT=55790TO55798:POKET,Y:NEXTT,I:REM 4E
115 DATA0,6,2,4,5,3,7,1,1,7,3,5,4,2,6,0:REM 7D
120 POKE53272,(PEEK(53272)AND15)OR160:REM 2C
130 PRINT"<CLR/HOME><HOME><GRN> |<30RIGHT>¯":REM 15
132 PRINT"¯<30RIGHT>¯<GREY2> SHIP:<GRN>"SH:REM 14
134 PRINT"<UP> K30RIGHT>,":REM FD
136 PRINT"o<30RIGHT>¯":REM 6D
138 PRINT"¯<30RIGHT>, <PUR>SCORE<GRN>":REM A1
140 PRINT"<14RIGHT>┌●┐ _●_<8RIGHT>x":REM F3
142 PRINT"/|<11RIGHT>♠| /////////¯<7RIGHT>¯"SC:REM 1
  4
144 PRINT"/¯<10RIGHT>♠/////////////|<6RIGHT>x":REM F
  E
146 PRINT"//●_<6RIGHT>┌┐V/////////////¯<6RIGHT>¯":RE
  M D5
148 PRINT"////_| ●_┌┐V///└┘└┘└/////<6RIGHT>¯ SPEED
  ":REM 5E
150 PRINT"//////////└\\| <5RIGHT>●└┘└┘| <6RIGHT>/":R
  EM 90
152 PRINT"//////\\| <20RIGHT>¯/   <RVS ON> <RVS OF
  F>":REM F3
154 PRINT"//////| <23RIGHT>¯/   <RVS ON> <RVS OFF>"
  :REM E5
156 PRINT"/////o<23RIGHT>♠//   <RVS ON> <RVS OFF>"
  :REM E6
158 PRINT"/////♠<21RIGHT>┌┐V///   <RVS ON> <RVS OFF
  >":REM 5B
160 PRINT"////o<6RIGHT>┌┐| __| ●_┌┐| _●┐┘| /////   <RV
  S ON> <RVS OFF>":REM 85
162 PRINT"////♠<5RIGHT>¯///<RVS ON>←THIS WAY,AL<RV
  S OFF>//////   <RED><RVS ON> <RVS OFF><GRN>":REM
   DO
164 PRINT"/// K6RIGHT>x└\\└└└└\\└\\└└└V/////   <RV
  S ON> <RVS OFF>":REM 94
166 PRINT"///<23RIGHT>●////   <RVS ON> <RVS OFF>"
  :REM EB
```

```
168 PRINT"///<24RIGHT>///   <RVS ON> <RVS OFF>":
  REM 3E
170 PRINT"////<23RIGHT>///   <RVS ON> <RVS OFF>"
  :REM E2
172 PRINT"////<24RIGHT>///   <RVS ON> <RVS OFF>":
  REM 33
174 PRINT"/////<6RIGHT>///   <RV
  S ON> <RVS OFF>":REM A5
176 PRINT"////////////////////////":REM 25
178 PRINT"////////////////////////<HOME>":
  REM 44
180 POKEV+21,5:POKEV+16,4:REM BE
200 POKE53272,(PEEK(53272)AND15)OR176:REM 32
205 A=48:S=80:Z=.5:X=0:POKEV+4,40:REM 94
210 POKE28664,240:POKE28665,241:POKE28666,248:REM
  04
215 POKEV+39,12:POKEV+40,1:POKEV+41,1:REM 9A
300 POKEV,A:POKEV+1,S:POKEV+2,A:POKEV+3,S+16:POKEV
  +5,168+(X*12):REM 21
310 X=X+GR:REM 45
350 IFPEEK(197)=4THENX=X-.2:SYS16000:REM 88
360 IFPEEK(197)=10THENZ=Z+.15:SYS16020:REM EE
370 IFPEEK(197)=18THENZ=Z-.15:SYS16020:REM F9
380 IFPEEK(197)>19THENSYS16040:REM 12
390 A=A+Z:S=S+X:REM 80
400 IFA>255ORS<30THENGOTO550:REM D3
410 IFPEEK(V+31)<>4THENGOTO510:REM 2B
500 GOTO300:REM 61
510 IFS>219THENGOTO900:REM 53
515 IFPEEK(V+31)=5THENA=A-Z:S=S-X:GOTO19000:REM 7B
516 IFPEEK(V+31)=7THENA=A-Z:S=S-X:GOTO19000:REM 7E
517 IFPEEK(V+31)=1THENA=A-Z:S=S-X:GOTO19000:REM 79
518 IFPEEK(V+31)=3THENA=A-Z:S=S-X:GOTO19000:REM 7C
520 GOTO500:REM 65
550 IFA>255THEN:A=255:GOTO19000:REM 34
560 IFS<30THENGOTO600:REM 1A
570 GOTO550:REM 6F
600 PRINT"<HOME><2DOWN><5RIGHT>WHERE ARE YOU GOING
  ?":POKEV+21,5:POKE54273,0:REM A8
```

```
640 FORT=1TO2000:NEXTT:PRINT"<HOME><2DOWN><5RIGHT>
  <20SPACE>":GOTO19100:REM B4
900 IFPEEK(V+5)>176THENGOTO19000:REM 92
905 POKE54273,0:POKE54276,17:POKEV+21,5:REM E1
909 BO=(177-PEEK(V+5))*50:REM 48
910 PRINT"<HOME><3DOWN><10RIGHT>BONUS<YEL>"BO:REM
  89
920 FORT=1TOBOSTEP10:POKE54273,100:FORB=1TO10:NEXT
  B:POKE54273,0:REM 9A
930 SC=SC+10:PRINT"<HOME><6DOWN><32RIGHT>"SC:NEXTT
  :REM 18
990 GOTO20000:REM CD
15000 POKEV,A:POKEV+1,S:POKEV+2,A:POKEV+3,S+19:POK
  EV+5,168+(X*10):REM 85
15005 IFPEEK(V+31)<>4THENGOTO18000:REM F4
15010 X=X+GR:A=A+Z:S=S+X:REM C6
15020 IFPEEK(197)=4THENX=X-.2:SYS16000:REM E8
15030 IFPEEK(197)=10THENZ=Z+.15:SYS16020:REM 4E
15040 IFPEEK(197)=18THENZ=Z-.15:SYS16020:REM 59
15050 IFPEEK(197)>19THENSYS16040:REM 72
15070 IFA>255ORA<10THEN16100:REM 4D
15100 W=W+1:POKE28676,Q(W):IFW=30THENW=0:REM 8C
16000 GOTO15000:REM 26
16010 REM  ** ONKO ULKONA KUVASTA **:REM A4
16100 IFA>255THENSC=SC+(LU*50):LU=LU+1:REM FB
16110 IFA<20THENSC=SC-(LU*50):LU=LU-1:REM C4
16150 ONLUGOTO20000,50000,60000,62000:REM CD
16200 END:REM DO
18000 IFPEEK(V+31)=5THEN19000:REM 7E
18005 IFPEEK(V+31)=1THEN19000:REM 7F
18010 IFPEEK(V+31)=7THEN19000:REM 81
18015 IFPEEK(V+31)=3THEN19000:REM 82
18020 GOTO15010:REM 2B
19000 POKEV+21,5:REM 6E
19001 POKE54277,12:POKE54278,0:POKE54294,50:POKEV+
  2,A:POKEV+3,S+2:POKE54273,2:REM E1
19002 IFLU=0THENPOKEV+2,A:POKEV+3,S:REM B3
19003 POKE28665,243:POKEV+21,7:POKEV,A:POKEV+1,S:R
  EM 72
```

```
19010 FORT=243TO246STEP.5:POKEV+40,5:FORG=1TO70:NE
  XTG:POKEV+40,2:REM E5
19015 FORG=1TO70:NEXTG:REM 66
19020 POKE28665,T:NEXTT:POKEV+21,6:REM 32
19030 FORT=246TO243STEP-.5:POKEV+40,5:FORG=1TO70:N
  EXTG:POKEV+40,2:REM 14
19035 FORG=1TO70:NEXTG:REM 68
19040 POKE28665,T:NEXTT:POKEV+21,0:REM 2E
19100 SH=SH-1:Z=0:X=0:IFLU=0THENGOTO19105:REM 0B
19101 IFC=1THENA=150:REM 7F
19102 IFC=0THENA=50:REM 4E
19105 IFSH=0THENGOTO19600:REM FF
19107 IFLU=0THENGOTO100:REM 98
19110 ONLUGOTO20000,50000,60000,62000:REM CC
19600 PRINT"<HOME><8DOWN><9RIGHT><13SPACE>":REM 71
19602 PRINT"<9RIGHT>  <PUR>G<GRN>A<CYN>M<YEL>E O<C
  YN>V<GRN>E<PUR>R  ":REM 7E
19603 PRINT"<9RIGHT><13SPACE>":REM D9
19605 FORT=1TO3000:NEXTT:GOTO1:REM 85
20000 LU=1:POKEV+40,1:POKE28665,241:POKEV+21,0:POK
  E54276,0:REM 59
20001 POKE54273,0:POKE54277,0:POKE54278,255:POKE54
  276,129:POKE54294,80:REM CA
20003 IFA<100THENA=249:S=180:C=0:GOTO20005:REM 95
20004 IFA>100THENA=70:S=70:C=1:REM C5
20005 POKE53272,(PEEK(53272)AND15)OR160:POKE28664,
  242:REM C7
20020 PRINT"<CLR/HOME><HOME><GRN>////////////////
  ////////////////":REM 49
20030 PRINT"////^L\\LL\\\/////////////////////<GREY2
  > SHIP:<GRN>"SH:REM 3A
20040 PRINT"<UP>////|<8RIGHT>*L\\L\L\LL\\L/////":
  REM 89
20050 PRINT"/// K22RIGHT>/////":REM 53
20060 PRINT"///\<22RIGHT>X/////<BLU> SCORE<GRN>":R
  EM 0F
20070 PRINT"///O<23RIGHT>////":REM B3
20080 PRINT"///♠<11RIGHT>♠—<10RIGHT>¯////"SC:REM 8
  2
```

```
20090 PRINT"//\<9RIGHT>┌┐└//●_<8RIGHT>`////":REM C
  4
20100 PRINT"//o<7RIGHT>┌┐/////////_|@@@@@@'////":REM
   4E
20110 PRINT"//-<7RIGHT>`/////////-<6RIGHT>,//// SP
  EED":REM E3
20120 PRINT"///-<6RIGHT>●└/////////\<6RIGHT>x////":
  REM 32
20130 PRINT"////●_<6RIGHT>,////////\<6RIGHT>¯////
   <RVS ON> <RVS OFF>":REM DE
20140 PRINT"//////_|<4RIGHT>x////////o<6RIGHT>x////
     <RVS ON> <RVS OFF>":REM 47
20150 PRINT"///////-<5RIGHT>●////└\♠@@@@@@@●///
  <RVS ON> <RVS OFF>":REM DC
20160 PRINT"////////-<5RIGHT>'//|<11RIGHT>●//   <R
  VS ON> <RVS OFF>":REM 1B
20170 PRINT"////////\<4RIGHT>¯//|<13RIGHT>x/   <RV
  S ON> <RVS OFF>":REM 6D
20180 PRINT"////////|<4RIGHT>¯/o<15RIGHT>,   <RED>
  <RVS ON> <RVS OFF><GRN>":REM 54
20190 PRINT"///////o<4RIGHT>♠//♠<16RIGHT>   <RVS O
  N> <RVS OFF>":REM 63
20200 PRINT"///////♠<4RIGHT>'/ K17RIGHT>   <RVS ON
  > <RVS OFF>":REM A5
20210 PRINT"//////|<5RIGHT>●/o<11RIGHT>┌_<4RIGHT>
    <RVS ON> <RVS OFF>":REM 6F
20220 PRINT"//////-<6RIGHT>x♠<5RIGHT>┌┐└| __//__●_
    <RVS ON> <RVS OFF>":REM 6F
20230 PRINT"///////-<11RIGHT>♠/////////////   <RVS
  ON> <RVS OFF>":REM 76
20240 PRINT"////////-<7RIGHT>┌┐|//////////////   <R
  VS ON> <RVS OFF>":REM 44
20250 PRINT"/////////●_<3RIGHT>┌┐//////////////////"
  :REM 81
20260 PRINT"////////////_||//////////////////////<HOME>
  ":REM 88
20270 POKE53272,(PEEK(53272)AND15)OR176:REM 9B
21000 POKEV+31,0:GOTO15000:REM CC
50000 LU=2:POKEV+40,1:POKE28665,241:POKEV+21,0:POK
  E54276,0:REM 5D
```

```
50001 POKE54273,0:POKE54277,0:POKE54278,255:POKE54
  276,129:POKE54294,80:REM CD
50003 IFA<100THENA=249:S=85:C=0:GOTO50005:REM 6F
50004 IFA>100THENA=21:S=181:C=1:REM F7
50005 POKE53272,(PEEK(53272)AND15)OR160:POKE28664,
  242:REM CA
50020 PRINT"<CLR/HOME><HOME><GRN>//////////////////
  ////////////////":REM 4C
50030 PRINT"//////////////////L\L\\////\LL\\L<GREY2
  > SHIP:<GRN>"SH:REM 38
50040 PRINT"<UP>//////////////////|<5RIGHT>,///o<6RIG
  HT>":REM 96
50050 PRINT"//////////////|<6RIGHT>x\//♠<6RIGHT>":
  REM 56
50060 PRINT"//////////////o<9RIGHT>,o<7RIGHT><BLU>
  SCORE<GRN>":REM 4E
50070 PRINT"//////////////♠<9RIGHT>x♠<7RIGHT>":REM
  99
50080 PRINT"////////////// K5RIGHT>┌_<11RIGHT>♠"SC:R
  EM 06
50090 PRINT"//////////////\<4RIGHT>♠/\<10RIGHT>¯/":R
  EM C1
50100 PRINT"//////////////o@@@@ \//_| _●_<5RIGHT>¯/":R
  EM C5
50110 PRINT"//////////////♠<4RIGHT>'////////_<4RIGHT>
  '/ SPEED":REM 67
50120 PRINT"////L\LL\\|<4RIGHT>¯////////\@@@@ \/":
  REM 77
50130 PRINT"////|<11RIGHT>¯////////// K4RIGHT> \/   <
  RVS ON> <RVS OFF>":REM C2
50140 PRINT"///|<12RIGHT>'////////\<4RIGHT>'/   <R
  VS ON> <RVS OFF>":REM 1A
50150 PRINT"//o<13RIGHT>,////////o<4RIGHT> \/   <RV
  S ON> <RVS OFF>":REM 84
50160 PRINT"//♠<13RIGHT>x\L/////\♠<4RIGHT> \/   <RV
  S ON> <RVS OFF>":REM 84
50170 PRINT"\|<5RIGHT>♠_●┐_<7RIGHT>♥///|<5RIGHT>¯/
  /   <RVS ON> <RVS OFF>":REM 59
50180 PRINT"<6RIGHT>¯/////|<7RIGHT>,/o<6RIGHT>¯//
    <RED><RVS ON> <RVS OFF><GRN>":REM 3B
```

```
50190 PRINT"<6RIGHT>¯/////¬<7RIGHT>×\♠<5RIGHT>♠///
     <RVS ON> <RVS OFF>":REM B2
50200 PRINT"<5RIGHT>♠///////┬_<11RIGHT>♠////   <R
  VS ON> <RVS OFF>":REM F9
50210 PRINT"<2RIGHT>┌┐_//////////●_<7RIGHT>┌┐////
  /   <RVS ON> <RVS OFF>":REM F9
50220 PRINT"_| ////////////////●_<2RIGHT>┌┐| ///////
     <RVS ON> <RVS OFF>":REM C0
50230 PRINT"//////////////////| _//////////   <RV
  S ON> <RVS OFF>":REM 20
50240 PRINT"//////////////////////////////////   <RV
  S ON> <RVS OFF>":REM 30
50250 PRINT"//////////////////////////////////":REM
  8D
50260 PRINT"//////////////////////////////////<HOME>
  ":REM A1
50270 POKE53272,(PEEK(53272)AND15)OR176:REM 9E
51000 POKEV+31,0:GOTO15000:REM CF
60000 LU=3:POKEV+40,1:POKE28665,241:POKEV+21,0:POK
  E54276,0:REM 5F
60001 POKE54273,0:POKE54277,0:POKE54278,255:POKE54
  276,129:POKE54294,80:REM CE
60004 IFA>100THENA=21:S=70:C=1:REM C5
60005 POKE53272,(PEEK(53272)AND15)OR160:POKE28664,
  242:REM CB
60020 PRINT"<CLR/HOME><HOME><GRN>//////////////////
  ////////////////":REM 4D
60030 PRINT"L\\\L\\/////////////////////////////<GREY2
  > SHIP:<GRN>"SH:REM 40
60040 PRINT"<UP><7RIGHT>●//////////////////////////"
  :REM 4A
60050 PRINT"<8RIGHT> ,//////////////////////////":REM
  0B
60060 PRINT"<8RIGHT>×///////L\\\/////////////<BLU>
  SCORE<GRN>":REM C1
60070 PRINT"<9RIGHT>'//\L\| <4RIGHT>●LL/////////":R
  EM 91
60080 PRINT"_| ___<4RIGHT> \/o<11RIGHT>●////////"SC:
  REM A3
```

```
60090 PRINT"////| <4RIGHT> \/♠<12RIGHT>♠///////":REM
   91
60100 PRINT"///| <4RIGHT>♠/O<14RIGHT> \//////":REM 1
  4
60110 PRINT"//O<4RIGHT>♠//♠<5RIGHT>|¯|| _|<4RIGHT>//
  ///// SPEED":REM E8
60120 PRINT"/♠<3RIGHT>¯// |@@@@♠_////<4RIGHT>////
  ///":REM 52
60130 PRINT"/ K4RIGHT>¯//\<4RIGHT>/////\<4RIGHT>\
  //////   <RVS ON> <RVS OFF>":REM C3
60140 PRINT"/\<4RIGHT>//O<4RIGHT>\///// K4RIGHT>/
  //////   <RVS ON> <RVS OFF>":REM D8
60150 PRINT"/ K4RIGHT>//♠<3RIGHT>¯//////\<4RIGHT>
  //////   <RVS ON> <RVS OFF>":REM 8F
60160 PRINT"/ K4RIGHT>♠\| <4RIGHT>¯///////_<3RIGHT>
  X//////   <RVS ON> <RVS OFF>":REM 8C
60170 PRINT"/\<10RIGHT>¯////////O<4RIGHT>/////
  <RVS ON> <RVS OFF>":REM 7C
60180 PRINT"//|<9RIGHT>¯////////♠@@@@/////   <RED
  ><RVS ON> <RVS OFF><GRN>":REM EF
60190 PRINT"//¯<8RIGHT>♠/////////| <4RIGHT>¯//////
   <RVS ON> <RVS OFF>":REM CD
60200 PRINT"///●_<4RIGHT>¬V////////\<5RIGHT>¯//<RV
  S ON>▬▬<3RIGHT> <RVS OFF>":REM CB
60210 PRINT"/////_| __/////////// K5RIGHT>♠\L<3RIGHT
  > ▌  <RVS ON> <RVS OFF>":REM 6B
60220 PRINT"////////////////////// K11RIGHT> ▌   <RVS
  ON> <RVS OFF>":REM 76
60230 PRINT"//////////////////////|<10RIGHT> ▌   <RVS
   ON> <RVS OFF>":REM 22
60240 PRINT"//////////////////////¯<10RIGHT> ▌   <RVS
   ON> <RVS OFF>":REM 24
60250 PRINT"//////////////////////| ___| | _<3RIGHT> ▛'
  :REM 2E
60260 PRINT"////////////////////////////<RVS ON>¯¯
  ¯¯<RVS OFF><HOME>":REM EA
60270 POKE53272,(PEEK(53272)AND15)OR176:REM 9F
61000 POKEV+31,0:GOTO15000:REM D0
62000 POKEV,252:POKEV+21,5:POKE28665,247:POKEV+40,
  0:POKEV+2,252:POKEV+3,201:REM 94
```

```
62001 V=53248:FORT=54272TO54296:POKET,0:NEXTT:POKE
  54296,15:POKE54277,202:REM 12
62005 POKEV+21,7:POKEV+23,2:POKE54278,0:POKE54276,
  17:REM F4
62010 RESTORE:FORT=1TO4:READY:NEXT:FORT=1TO4:READY
  :POKEV+39,Y:FORG=1TO100:NEXTG:REM 0D
62011 NEXTT:REM 8D
62015 RESTORE:FORT=1TO8:READY:POKEV+40,Y:FORG=150T
  00STEP-20:POKE54273,G:NEXTG,T:REM 5C
62020 FORT=1TO8:READY:POKEV+39,Y:POKEV+40,Y:FORG=1
  50TO0STEP-20:POKE54273,G:NEXTG:REM 55
62025 NEXTT:POKEV+21,0:POKEV+23,0:POKE54273,0:LE=L
  E+1:REM A5
62027 PRINT"<CLR/HOME><PUR><9DOWN><10RIGHT>LEVEL<G
  RN>"LE"<PUR>COMPLETED":REM 60
62029 PRINT"<DOWN><12RIGHT>BONUS"(SH*100)+(LE*50):
  REM 0B
62030 FORY=1TO8:RESTORE:FORT=1TO8:READI:FORG=55754
  TO55758:POKEG,I:NEXTG,T,Y:REM 89
62040 SC=SC+(SH*100)+(LE*50):GR=GR+.007:IFGR>.15TH
  ENGR=.15:REM 39
62050 PRINT"<CLR/HOME>":GOTO100:REM 65
```

Katko

Jyrki J. J. Kasvi

Katko — tunnetaan myös nimellä 'Viimeinen tikki' — on säännöiltään kenties yksinkertaisin olemassa oleva korttipeli. Tarkoituksena on saada viimeinen tikki viidesti ennen vastustajaa.

Aloitusvuorot vuorottelevat ja kiinniotto tapahtuu lyömällä suurempi samaa maata oleva kortti kuin aloittaja. Jos näin ei käy — on peli yhä aloittajalla kiinni, ja tämä jatkaa lyömistä ja niin edelleen. Altavastaajan tulee lyödä — jos mahdollista — samaa maata kuin kiinnipitäjä; altavastaajan ei kuitenkaan tarvitse lyödä suurempaa korttia samaa maata kuin kiinnipitäjä (eli ottaa kiinni), vaan pienempikin kortti samaa maata kelpaa.

Se, jolla peli viimeisen tikin jälkeen on kiinni, saa pisteen, ja se, joka saa ensiksi viisi pistettä, voittaa pelin.

Se voittaja et kuitenkaan usein ole Sinä!

Ohjeet koneelle annetaan painamalla kyseistä korttia vastaavaa numeroa (1-5). Pelin niksit oppii vain pelaamalla, joskin ohjelma on juonikas vastustaja, joka tuntuu piilottelevan ässiä ylimääräisissä muistiosoitteissaan.

PROGRAM:KATKO

```
10 REM                    **** KATKO ****:REM OF
15 ::REM AO
20 REM WRITTEN BY JYRKI J. J. KASVI FOR MIKROBITTI
  :REM EA
25 ::REM A1
50 DIM PA$(52): KO$(0)="15":REM OC
60 POKE53248+32,1:POKE53248+33,1:PRINT"<BLU>":REM
  ED
70 GOSUB 5000: REM OHJEET:REM 89
100 GOSUB 1000: REM PAKAN LUKU:REM 9C
110 PRINT"<DOWN>ALOITATKO <RVS ON>K<RVS OFF>/<RVS
  ON>E<RVS OFF><2UP>":GETAL$:IFAL$<>"K"ANDAL$<>"E"
  THEN110:REM DB
120 K1=1:KI=0: PI=0: K=1:REM EB
```

```
130 GOSUB 2000: REM JAKO:REM 19
140 PRINT"<CLR/HOME>":REM D7
150 PRINT"<HOME>SINUN PISTEESI";PI:REM 41
160 PRINT"C-64:N PISTEET";KI:REM B7
170 IFKI=50RPI=5THEN6000: VOITON MAARITTELY:REM E3
175 REM TILANNE POYDALLA:REM E2
180 PRINT"<BLU><2DOWN>MITA LYOT?<DOWN>":REM 6E
190 PRINT"<5SPACE> ┌──┐ ┌──┐ ┌──┐ ┌──┐ ┌──┐":R
  EM C1
200 PRINT"<5SPACE>|"PO$(1)"| |"PO$(2)"| |"PO$(3)"|
    |"PO$(4)"| |"PO$(5)"|":REM 9A
210 PRINT"<5SPACE> └──┘ └──┘ └──┘ └──┘ └──┘<DO
  WN>":GOSUB3000:REM 1B
220 PRINT"<BLU><DOWN><5SPACE> ┌──┐ ┌──┐ ┌──┐ ┌─
  ─┐ ┌──┐":REM EB
230 PRINT"<5SPACE>|"KA$(1)"<BLU>| |"KA$(2)"<BLU>|
   |"KA$(3)"<BLU>| |"KA$(4)"<BLU>| |"KA$(5)"<BLU>|"
  :REM D9
240 PRINT"<5SPACE> └──┘ └──┘ └──┘ └──┘ └──┘<DO
  WN>":REM A1
250 PRINT"KIERROS";K:IFAL$="K"THENPRINT"SINULLA KI
  INNI ":REM 2D
255 IFAL$="E"THENPRINT"C-64:LLA KIINNI":REM 51
258 IFK>5THEN PRINT"MIELENKIINTOISTA":REM A5
260 IFK>7THEN PRINT"<UP>KIPERA PAIKKA   ":REM E5
265 IFK>8THEN PRINT"<UP>TODELLA TIUKKAA ":REM 8D
270 IFAL$="E"THEN 400:REM 62
280 REM PELAAJA LYO"":REM 82
290 GETL$:IFL$<>"1"ANDL$<>"2"ANDL$<>"3"ANDL$<>"4"A
  NDL$<>"5"THEN290:REM AF
300 L=VAL(L$):REM CO
310 IFKA$(L)="   "THEN290:REM BB
315 FU=0:IFAL$="E"THEN GOSUB 1500: REM FUSKUAKO?:R
  EM 1C
317 IFFU=1THEN290:REM FD
320 PL$(K1)=KA$(L):GOSUB3000:REM 29
330 IFAL$="E"THEN700:REM 62
400 REM KUUSNELONEN LYO:REM C3
405 GOSUB3000:REM DC
410 IFAL$="K"THEN500:REM 65
```

```
420 LA=15:FORA=1TO5:R=VAL(RIGHT$(KO$(A),2)):REM 0C
430 IFR<LATHENLA=R:LY=A:REM E9
440 NEXT:KL$(K1)=KO$(LY):GOSUB3000:REM 07
450 IFAL$="K"THEN 700:REM 6B
460 GOTO 280:REM 6D
500 LY=0:FORA=1TO5:IFMID$(KO$(A),2,1)<>MID$(KA$(L)
  ,2,1)THEN 520:REM 09
510 IFRIGHT$(KO$(A),2)<RIGHT$(KO$(LY),2)THENLY=A:R
  EM 59
520 NEXT:IFLY=0THEN420:REM 76
530 KL$(K1)=KO$(LY):GOSUB3000:REM 8E
700 REM KELLA KIINNI?:REM E5
710 B=0:REM 47
720 IFAL$="K"ANDMID$(KA$(L),2,1)=MID$(KO$(LY),2,1)
  THENB=.5:REM DB
730 IFRIGHT$(KO$(LY),2)>RIGHT$(KA$(L),2)THENB=B+.5
  :REM 88
740 IFAL$="E"ANDMID$(KA$(L),2,1)<>MID$(KO$(LY),2,1
  )THENB=1:REM E2
750 IFAL$="E"ANDRIGHT$(KA$(L),2)<RIGHT$(KO$(LY),2)
  THENB=1:REM 33
760 PL$(K1)=KA$(L):KL$(K1)=KO$(LY):KO$(LY)="16":KA
  $(L)="    ":PO$(LY)="    ":REM B9
770 K1=K1+1:IFK1=6THEN800:REM AE
780 IFB=1THENAL$="E":GOTO 150:REM 8D
790 AL$="K":GOTO 150:REM 26
800 FORVI=1TO2000:NEXT:K1=1:K=K+1:REM F7
805 FORA=1TO5:PL$(A)="    ":KL$(A)="    ":NEXT:REM 9
  9
810 IFB=1THENKI=KI+1:GOTO830:REM D6
820 PI=PI+1:REM 65
830 IFQ=1THENQ=0:AL$="E":GOTO 130:REM 8E
840 Q=1:AL$="K":GOTO 130:REM 19
1000 REM PAKAN LUKU:REM 51
1010 FORA=1TO26: READA$: PA$(A)="<BLK>"+A$: NEXT:R
  EM B9
1020 FORA=27TO52: READA$: PA$(A)="<RED>"+A$: NEXT:
  REM 7D
1030 RETURN:REM A4
```

```
1500 REM FUSKUAKO?:REM 52
1510 IFMID$(KA$(L),2,1)=MID$(KO$(LY),2,1)THEN RETU
  RN:REM B7
1520 FORA=1TO5:REM 36
1530 IFMID$(KA$(A),2,1)=MID$(KO$(LY),2,1)THEN FU=1
  :REM D7
1540 NEXT:REM 09
1550 RETURN:REM AB
2000 REM JAKO:REM CB
2010 FORA=1TO5:REM 31
2020 KA$(A)=PA$(52*RND(1)+1):REM 9C
2030 FORT=0TOA-1:REM AF
2040 IFKA$(T)=KA$(A)THEN2020:REM 1C
2050 NEXT T,A:REM C7
2060 FORA=1TO5:REM 36
2070 KO$(A)=PA$(52*RND(1)+1):REM AF
2080 FORT=0TOA-1:REM B4
2090 IFKO$(T)=KO$(A)THEN2070:REM 42
2100 NEXT T:REM 56
2110 FORT=1TO5:REM 45
2120 IFKA$(T)=KO$(A)THEN2070:REM 2E
2130 PO$(A)="▒▒▒▒":REM 8E
2140 NEXT T,A:REM C7
2150 RETURN:REM A8
3000 REM TILANNE POYDALLA:REM 08
3010 PRINT"<HOME><11DOWN><3RIGHT>";:FORA=1TO5:REM
  9D
3020 PRINT"    "KL$(A);:REM 1E
3030 NEXT:PRINT:PRINT"<DOWN><3RIGHT>";:REM 7A
3040 FORA=1TO5:REM 35
3050 PRINT"    "PL$(A);:REM 26
3060 NEXT:PRINT"<DOWN>":REM 24
3070 RETURN:REM AA
4000 DATA"♠14","♠ 2","♠ 3","♠ 4","♠ 5","♠ 6","♠ 7"
  ,"♠ 8","♠ 9":REM 7C
4010 DATA"♠10","♠11","♠12","♠13":REM 01
4020 DATA"♠14","♠ 2","♠ 3","♠ 4","♠ 5","♠ 6","♠ 7"
  ,"♠ 8","♠ 9":REM 4D
4030 DATA"♠10","♠11","♠12","♠13":REM 5F
```

```
4040 DATA"♥14","♥ 2","♥ 3","♥ 4","♥ 5","♥ 6","♥ 7"
  ,"♥ 8","♥ 9":REM 22
4050 DATA"♥10","♥11","♥12","♥13":REM 4D
4060 DATA"♦14","♦ 2","♦ 3","♦ 4","♦ 5","♦ 6","♦ 7"
  ,"♦ 8","♦ 9":REM 63
4070 DATA"♦10","♦11","♦12","♦13":REM 6B
5000 REM OHJEET:REM 68
5010 PRINT"<CLR/HOME><16RIGHT><DOWN>OHJEET!<2DOWN>
  ":REM OD
5020 PRINT"   KATKO ON HYVIN YKSINKERTAINEN PELI."
  :REM A2
5030 PRINT"   SIINA ON TARKOITUKSENA SAADA VIISI":
  REM 4B
5040 PRINT"  VIIMEISTA TIKKIA ENNEN VASTAPELAAJAA.
  ":REM 30
5050 PRINT"  VIIMEISEN TIKIN SAA SE, JOLLA ON PELI
  ":REM 73
5060 PRINT"<5SPACE>VIIMEISEN VUORON JALKEEN KIINNI
  .":REM F6
5070 PRINT"<DOWN> PELISSA ON MAANTUNNUSTUSPAKKO, M
  UTTA EI":REM FB
5080 PRINT"<UP>  YLIOTTOPAKKOA. ALTAVASTAAJA SAA S
  IIS":REM A0
5090 PRINT"   PELIN KIINNI  LYOMALLA SAMAA MAATA":
  REM D9
5100 PRINT"   SUUREMMAN KORTIN KUIN KIINNIPITAJA."
  :REM E9
5110 PRINT"   KORTIN EI KUITENKAAN TARVITSE OLLA":
  REM 60
5120 PRINT"   SUUREMPI, VAIKKA SELLAINEN KADESSA":
  REM 6B
5130 PRINT"   OLISIKIN. RIITTAA, ETTA SE ON SAMAA"
  :REM 08
5140 PRINT"<17SPACE>MAATA.":REM 0B
5150 PRINT"<DOWN>   LYOTAVAT KORTIT MAARATAAN NUME
  RO-":REM 65
5160 PRINT"  NAPPAIMILLA 1-5 VASEMMALTA OIKEALLE."
  :REM 95
5200 RETURN:REM A7
```

```
6000 REM VOITON MAARITTELY:REM 85
6010 IFKI=5THEN6100:REM 52
6020 IFPI-KI=1THEN6070:REM 1B
6030 IFPI-KI<3THEN6050:REM 1B
6040 PRINT"<2DOWN>SILKKAA SATTUMAA!!!":GOTO 6500:R
  EM BE
6050 PRINT"<2DOWN>VOIHAN KULMAKARVAT":GOTO 6500:RE
  M 04
6070 PRINT"<2DOWN>VAADIN UUSINTAOTTELUA":FORVI=1TO
  3000:NEXT:GOTO 110:REM CC
6100 IFKI-PI>3THEN6120:REM 19
6110 PRINT"<2DOWN>KIITOS KOVASTA KISASTA":GOTO6500
  :REM F5
6120 PRINT"<2DOWN>HAHHAHHAAA!!!!!!!!!!!!!!":REM 37
6500 PRINT"<2DOWN>HALUATKO UUSINTAA <RVS ON>K<RVS
  OFF>/<RVS ON>E<RVS OFF><DOWN>":REM 99
6510 GETU$:IFU$<>"K"ANDU$<>"E"THEN6510:REM BA
6520 IFU$="K"THEN110:REM 62
6530 PRINT"<DOWN>OLI ILO(?!!??) PELATA KANSSASI";:
  REM 17
6540 GOTO6540:REM D7
```

Jousiammunta

Harri Holopainen

Jousella ampumista on mukava harrastaa Kuusnelosenkin kanssa. Mukavasti tämä käy päinsä oheisen ohjelman avulla.

Pelissä on tarkoituksena arvata oikea laukaisuhetki välilyöntiä painamalla. Ohjelma ei ole kuitenkaan aivan niin yksinkertainen, kuin mitä tästä voisi päätellä, sillä ampumapaikalla puhaltaa myös varsin navakka tuuli. Tuulen voimakkuus ja suunta käyvät ilmi näytön ylälaidassa olevasta laatikosta.

Peli alkaa F1:tä painamalla ja peliohjeet saa esiin F3:n avulla. Pelattaessa välilyönnin painallus ilmoittaa laukaisuhetken.

Alussa ohjelma arpoo tuulen suunnan ja voimakkuuden. Tämän ja muiden alkurutiinien jälkeen siirytään maalitaulun siirtorutiineihin. Riveille 500-620 siirretään pelkkää maalitaulua ja tarkistetaan, onko nuoli jo laukaistu. Riveillä 700-713 liikutetaan sekä nuolta että taulua. Jos nuoli osuu, siirretään sekä taulu, että siihen osunut nuoli alas riveillä 713-715. Mikäli osumaa ei tullut, liikutetaan taulu pois näytöltä rivillä 720.

Osuman jälkeen ohjelma sijoittaa näytön alanurkassa sijaitsevaan tauluun nuolen (rivit 900-910) sekä tulostaa tuloksen (rivit 740-755 ja 2200-2220).

PROGRAM: JOUSIAMMUNTA

```
1 FORN=0TO62:READQ:POKE12800+N,Q:NEXT:REM 58
2 FORN=0TO62:READQ:POKE12864+N,Q:NEXT:REM 63
3 FORN=0TO62:READQ:POKE12928+N,Q:NEXT:REM 65
10 POKE53280,11:POKE53281,5:V=53248:POKE2040,200:P
   OKE2041,202:POKE2042,201:REM 8D
11 FORN=2043TO2047:POKEN,201:NEXT:FORN=6TO14STEP2:
   POKEV+N,0:NEXT:REM 61
12 FORN=42TO46:POKEV+N,0:NEXT:REM 55
13 TA$="<BLU><RVS ON> <RVS OFF><LEFT><DOWN><RVS ON
   > <RVS OFF><LEFT><DOWN><RVS ON> <RVS OFF><LEFT><
   DOWN><RVS ON> <RVS OFF><LEFT><DOWN><RVS ON> <RVS
    OFF><LEFT><DOWN><RVS ON> <RVS OFF><UP><BLK>+<LE
   FT><3UP>+<BLK>":REM 62
```

```
15 POKEV+16,0:NU$="↑↑↑↑↑↑":F=0:GOSUB1000:REM 99
20 PM=0:Y=125:T=SGN(RND(1)*10-5):W=RND(1)/(6-VT):R
   EM EC
22 TU=INT(W*100):IFT=1THENT$="▼<2LEFT><UP>▄<DOWN
   >":REM 16
23 PRINT"<HOME><DOWN><12RIGHT><5SPACE><5RIGHT><9SP
   ACE>":IFT=-1THENT$="<UP><RVS ON>▛▜<RVS OFF><2LEF
   T><DOWN>▀":REM A9
24 PRINT"<HOME><DOWN> TUULI:   ";T$;"   "TU"M/S<6SPA
   CE>";NU$:REM 18
25 PRINT"────────────────────────────────────────/<UP><LEFT
   >|<UP><LEFT>|<UP><LEFT>|<HOME><15DOWN><4RIGHT>"T
   A$:REM 0C
26 POKEV+4,0:REM A8
30 FORN=1060TO2020STEP40:POKEN,118:POKEN+54272,14:
   NEXT:REM 10
50 POKEV+16,1:POKEV,40:POKEV+2,50:POKEV+3,125:REM
   B6
60 GOTO500:REM 34
200 POKES+1,30:POKES+4,17:FORN=0TO10:NEXT:POKES+4,
    16:RETURN                          :REM CF
210 POKES+1,7:POKES+4,129:RETURN        :REM D1
220 POKES+1,15:POKES+4,17:FORN=0TO700:NEXT:POKES+4
    ,0:FORN=0TO200:NEXT:POKES+4,17:REM F2
225 POKES+1,10:FORN=0TO800:NEXT:POKES+4,0:RETURN:R
    EM 25
250 GOSUB200:GOSUB200:GOSUB200:FORN=0TO90:NEXT:GOS
    UB200:FORN=0TO90:NEXT:GOSUB200:REM 9B
252 RETURN:REM 79
499 REM*** LENKKI 1 ***:REM 75
500 FORQ=0TO255STEP.7:POKEV+1,Q:REM B2
510 A%=PEEK(197):REM 50
520 IFA%=60THEN700:REM F5
600 NEXT:REM D5
620 GOTO500:REM 66
699 REM*** LENKIT 2-4 ***:REM E2
700 FORL=50TO350STEP7:POKEV+5,Y:Y=Y+T*W:REM E7
701 IFL>250THENPOKEV+16,5:POKEV+4,L-250:GOTO710:RE
    M 54
```

```
702 POKEV+4,L:REM F5
710 Q=Q+1.7:POKEV+1,Q:REM D0
712 IFL>287THENIFY>Q-10.5THENIFY<Q+10.5THEN720
                                :REM 46
713 NEXT:GOSUB210:FORJ=QTO255STEP.6:POKEV+1,J:IFJ>
  Q+20THENPOKES+4,0:REM 2B
714 NEXT:GOTO725:REM EC
720 GOSUB200:E=Q-Y:FORJ=QTO240STEP.7:POKEV+1,J:POK
  EV+5,J-E:NEXT:GOSUB900:REM 75
725 POKEV+16,1:POKEV+4,0:REM 8C
729 REM*** PISTELASKU ***:REM 87
730 NU$=LEFT$(NU$,LEN(NU$)-1):L%=0:REM 9E
735 Y2=INT(Y):N2=INT(Q):REM 80
740 FORP=0TO10:IFN2+P-10=Y2THENPM=10*P+10:REM CD
745 NEXT:REM DF
750 FORP=0TO10:IFN2-P+10=Y2THENPM=10*P+10:REM CE
755 NEXT:REM E0
760 GOSUB2200:IFNU$=""THEN790:REM C1
770 GETH$:IFH$=""THEN770:REM 6D
780 GOTO20:REM 3A
789 REM*** LOPPURUTIINIT ***:REM 90
790 PRINT"<HOME><10DOWN><9SPACE><RVS ON>  **  GAME
   OVER  **  ":GOSUB220:REM 8C
800 FORM=0TO2000:NEXT:REM 17
805 PM=0:POKEV+21,0:PRINT"<CLR/HOME>KOKONAISPISTEE
  SI:";KP;" TASOLLA ";VT:POKE53280,5:REM 03
810 IFKP>HSTHENPRINT"<DOWN><WHT>ONNEA!<BLK>  TULOK
  SESI ON PAIVAN PARAS!":GOSUB250:REM DC
820 IFKP>HSTHENINPUT"<DOWN>NIMIKIRJAIMESI...";NI$:
  HS=KP:V0=VT:GOTO840:REM C8
830 PRINT"<DOWN>PAIVAN PARAS TULOS:":PRINT"<DOWN>"
  NI$": "HS" PISTETTA TASOLLA "V0:REM DB
840 FORN=0TO8000:NEXT:KP=0:GOTO10:REM 38
900 Y3=(10.5-(Q-Y))*2.3+160:POKEV+6+F,40:POKEV+7+F
  ,Y3:F=F+2:REM 16
910 RETURN:REM 7A
999 REM*** ALKURUTIINIT ***:REM 30
1000 POKEV+21,255:POKEV+28,253:POKEV+37,0:POKEV+38
  ,7:POKEV+40,1:PRINT"<BLK><CLR/HOME>":REM F7
```

```
1001 POKEV+27,250:POKE53281,11:POKEV+39,13:POKEV+4
  1,6:REM 2F
1002 S=54272:POKES,100:POKES+5,0:POKES+6,240:POKES
  +24,15:REM 06
1010 PRINT" <LT.BLUE><RVS ON><38SPACE>":REM 3F
1020 PRINT" <RVS ON> <RVS OFF><36SPACE><RVS ON> <R
  VS OFF>":REM DC
1030 PRINT" <RVS ON> <RVS OFF><BLK><10SPACE>JOUSIA
  MMUNTA<14SPACE><LT.BLUE><RVS ON> <RVS OFF>":REM
  A4
1040 PRINT" <RVS ON> <RVS OFF><36SPACE><RVS ON> <R
  VS OFF>":REM DE
1050 PRINT" <RVS ON><38SPACE>":REM A9
1060 PRINT"<BLK><HOME><15DOWN><7SPACE>**** ALOITA
  F1:LLA  ****":REM CE
1061 PRINT"<DOWN><7SPACE>**** OHJEITA F3:LLA ****"
  :REM 8A
1065 PRINT"<LT.BLUE><4DOWN> ****  PROGRAM BY HARRI
   HOLOPAINEN ****":REM E1
1066 PRINT"<LT.RED> ** COPYRIGHT HARRI HOLOPAINEN
  1985  **":REM CF
1067 POKEV+2,78:POKEV+3,125:REM 2C
1070 POKEV,250:POKEV+1,125:POKEV+5,125:FORN=78TO24
  5STEP2:POKEV+4,N:REM 31
1100 GETA$:IFA$="<F1>"THENGOSUB3100:GOTO1120:REM E
  5
1105 IFA$="<F3>"THENGOSUB3000:REM 34
1107 NEXT:REM 08
1110 GOTO1070:REM C4
1120 PRINT"<CLR/HOME><BLK>":POKEV,0:POKEV+4,0:POKE
  53281,11:REM 47
1130 FORN=1544TO1551:POKEN,160:POKEN+54272,11:NEXT
  :REM DB
1140 FORN=1551TO1991STEP40:POKEN,160:POKEN+54272,1
  1:NEXT:REM 82
2200 PRINT"<HOME><21DOWN><10RIGHT><4SPACE>PISTEET:
   ":KP=KP+PM          :REM FC
2210 PRINT"<6SPACE><4LEFT><WHT>"PM"<BLK><4RIGHT><1
  1SPACE><7LEFT>"KP:REM DA
```

```
2220 POKE53281,5:RETURN:REM 73
3000 REM*** OHJEET ***:REM 62
3010 PRINT"<CLR/HOME><LT.GREEN><DOWN>AUTA JOUSIAMM
  UNTAKILPAILUIHIN JOUTUNUT-":REM D8
3020 PRINT"<DOWN>TA JASKAA AMPUMAAN VIISI NUOLTA."
  :REM BC
3030 PRINT"<DOWN>KILPAILUPAIKALLA KUITENKIN TUULEE
  ,":REM F4
3040 PRINT"<DOWN>JOTEN OLE TARKKANA.":REM 84
3050 PRINT"<9DOWN>NUOLEN LAUKAISET VALILYONNILLA."
  :REM B2
3060 PRINT"<DOWN>ONNEA KISAAN!":FORN=0TO5000:NEXT:
  GOTO10:REM 85
3099 REM*** VAIK. TASO ***:REM 45
3100 INPUT"<CLR/HOME><WHT>VAIKEUSTASO (1-5) ";VT:I
  FVT>5ORVT<1THEN3100:REM D9
3110 RETURN:REM A5
10000 DATA0,0,160,0,0,160,0,0,160:REM 50
10010 DATA0,0,160,0,0,160,0,0,172:REM 54
10020 DATA0,0,160,0,0,160,0,0,160:REM 52
10030 DATA0,0,160,0,0,160,0,0,160:REM 53
10040 DATA0,0,160,0,0,160,0,0,160:REM 54
10050 DATA0,0,172,0,0,160,0,0,160:REM 58
10060 DATA0,0,160,0,0,160,0,0,160:REM 56
10070 REM*******************************:REM 9E
10100 DATA0,0,0,0,0,0,0,0,0:REM 1C
10110 DATA0,0,0,0,0,0,0,0,0:REM 1D
10120 DATA0,0,0,0,0,0,0,0,0:REM 1E
10130 DATA10,0,0,2,255,244,10,0,0:REM 59
10140 DATA0,0,0,0,0,0,0,0,0:REM 20
10150 DATA0,0,0,0,0,0,0,0,0:REM 21
10160 DATA0,0,0,0,0,0,0,0,0:REM 22
10170 REM*******************************:REM 9F
10200 DATA0,0,0,0,0,0,0,0,0:REM 1D
10210 DATA0,0,0,0,48,0,3,136,0:REM C7
10220 DATA27,180,0,15,228,0,15,68,0:REM D7
10230 DATA23,226,0,38,102,0,127,255,128:REM 9E
10240 DATA59,178,0,12,100,0,15,196,0:REM 09
10250 DATA15,196,0,15,200,0,15,176,0:REM 04
10260 DATA12,192,0,12,192,0,14,224,0:REM FE
```

Formula

Vesa Takkinen

Olet lähtenyt vanhalla kuhmuisella autollasi jokamiesluokan kilpa-ajoihin, mutta eksyit matkalla ja päädyitkin Formula-1 -radalle.

Lähtövalot syttyvät, ja ajokkisi syöksähtää liikkeelle. Mutta mitäs nyt!? Oletkin vahingossa lähtenyt väärään suuntaan! Ainoa mahdollisuutesi on väistellä vastaantulevia formuloita niin pitkään kuin pystyt. Aluksi tämä tuntuu helpolta, mutta kunhan autosi kiihtyy täyteen vauhtiinsa, niin helppous on kaukana.

Ohjelma jakaantuu seuraaviin perusosiin:

0 —	335	Alustukset
1000 —	1180	Alkuruutu
2000 —	2130	Pääohjelman Basic-osa
2145 —	2240	Pisteiden tulostus
10000 —	30000	Aliohjelmat
60000 —	62011	Datat

Ohjelmassa spritejen liikuttelu on toteutettu konekielellä, sillä Basic on kahdeksan spriten portaattomaan liikutteluun aivan liian hidas. DATA-lauseiden kirjoittamisessa kannattaa olla erityisen huolellinen, jottei virheitä syntyisi, joskin tarkastaja-ohjelman käyttö helpottaa virheiden selvittämistä suuresti.

```
PROGRAM:FORMULA

0 S=54272:FORI=STOS+24:POKEI,0:NEXT:POKES+24,15:PR
  INT"<CLR/HOME>":GOSUB30000:REM F6
1 DIMH%(2,200),L%(2,200),C%(2,200),FQ(11):FORI=1TO
  7:READX:NEXT:POKEV+21,0:REM 64
2 V(0)=33:V(1)=17:V(2)=17:FORI=0TO11:READFQ(I):NEX
  T:FORK=0TO2:REM A5
3 I=0:REM E9
4 READNM:CH=CH+NM:IFNM=0THEN14:REM A3
5 WA=V(K):WB=WA-1:IFNM<0THENNM=-NM:WA=0:WB=0:REM 2
  0
6 DR%=NM/128:OC%=(NM-128*DR%)/16:NT=NM-128*DR%-16*
  OC%:FR=FQ(NT):REM 5F
```

```
7 IFOC%=7THEN9:REM 59
8 FORJ=6TOOC%STEP-1:FR=FR/2:NEXT:REM 51
9 HF%=FR/256:LF%=FR-256*HF%:REM FA
10 IFDR%=1THENH%(K,I)=HF%:L%(K,I)=LF%:C%(K,I)=WA:I
  =I+1:GOTO4:REM FA
11 FORJ=1TODR%-1:H%(K,I)=HF%:L%(K,I)=LF%:C%(K,I)=W
  A:I=I+1:NEXT:REM 7B
12 H%(K,I)=HF%:L%(K,I)=LF%:C%(K,I)=WB:REM 0A
13 I=I+1:GOTO4:REM 36
14 IFI>IMTHENIM=I:REM 5C
15 NEXT:REM A5
19 V=53248:REM 03
25 IFPEEK(828)=166THEN100:REM A8
30 SP=16368:REM 4B
35 READX:CH=CH+X:IFX=-1THEN50:REM 3C
40 POKESP,X:SP=SP+1:GOTO35:REM AE
50 SP=832:REM E2
55 READX:CH=CH+X:IFX=-1THEN100:REM 6A
60 POKESP,X:SP=SP+1:GOTO55:REM B2
100 POKE53276,255:POKEV+37,244:POKEV+38,240:REM C0
102 POKE2041,13:POKE2042,14:POKE2043,13:POKE2044,1
  4:POKE2045,13:POKE2046,14:REM EB
103 POKE2047,13:REM 20
110 POKEV+39,2:POKE5*4096,0:POKE6*4096,0:POKE49664
  ,0:POKE16473,10:POKE49408,0:REM 96
115 RESTORE:REM BB
120 FORI=3TO15STEP2:READHU:CH=CH+HU:POKEI+V,HU:NEX
  T:REM 67
125 IFPEEK(828)=166THEN320:REM DD
130 IFCH<>112159THENPRINT"<CLR/HOME><RVS ON>VIRHE
  - TARKISTA DATA-LAUSEET!!!":END:REM DC
320 POKES+5,0:POKES+6,250:POKES+12,15:POKES+13,245
  :POKES+19,15:POKES+20,245:REM 94
335 POKE828,166:REM 35
1000 POKE53280,0:POKE53281,0:PRINT"<CLR/HOME>":REM
   92
1010 TX$="*FORMULA-1*FORMULA-1*FORMULA-1*FORMULA-1
  *FORMULA-1*":REM 53
1020 PRINT"<HOME><5DOWN><UPPER/LOWER OFF>"CHR$(142
  ):REM ED
```

```
1030 PRINT"<LT.BLUE><11RIGHT>_______<10RIGHT>__":RE
  M BO
1040 PRINT"<LT.BLUE><10RIGHT>/<5RIGHT>/| <8RIGHT>/<
  RIGHT>/| ":REM 6A
1050 PRINT"<RED><RVS ON><10RIGHT><6SPACE><RVS OFF>
  <LT.BLUE>/<8RIGHT><RED><RVS ON>▛  <LT.BLUE><RVS
  OFF><RIGHT>| ":REM 8E
1060 PRINT"<RED><RVS ON><10RIGHT>  <RVS OFF><LT.BL
  UE><RIGHT>|___<4RIGHT>__<2RIGHT><RED><RVS ON>▛
  <RVS OFF><LT.BLUE><RIGHT>| ":REM 2C
1070 PRINT"<RED><RVS ON><10RIGHT>  <RVS OFF><LT.BL
  UE>/<2RIGHT>/| <2RIGHT>/<RIGHT>/| <3RIGHT><RED><RV
  S ON>  <RVS OFF><LT.BLUE><RIGHT>| ":REM 9B
1080 PRINT"<RED><RVS ON><10RIGHT><5SPACE><RVS OFF>
  <LT.BLUE>/<3RIGHT><RED><RVS ON>  <RVS OFF><LT.BL
  UE><RIGHT>| <3RIGHT><RED><RVS ON>  <LT.BLUE><RVS
  OFF><RIGHT>| ":REM CA
1090 PRINT"<RED><RVS ON><10RIGHT>  <RVS OFF><LT.BL
  UE><RIGHT>| <5RIGHT><RED><RVS ON>  <RVS OFF><LT.B
  LUE>/<3RIGHT>_<RED><RVS ON>  <RVS OFF><LT.BLUE><
  RIGHT>|__":REM 8A
1100 PRINT"<RED><RVS ON><10RIGHT>  <RVS OFF><LT.BL
  UE><RIGHT>| <10RIGHT>/<RED><RVS ON><RIGHT>  <RVS
  OFF><LT.BLUE>/ /| ":REM 79
1110 PRINT"<RED><RVS ON><10RIGHT>  <RVS OFF><LT.BL
  UE>/<11RIGHT><RED><RVS ON><6SPACE><RVS OFF><LT.B
  LUE>/":REM 45
1120 PRINT"<3DOWN><15RIGHT>COPYRIGHT 1985":REM 0B
1122 PRINT"<2UP><11RIGHT> ,-.<DOWN><3LEFT> |C |<DOWN>
  <3LEFT> `-' ":REM 48
1125 PRINT"<DOWN><8RIGHT><RVS ON>Z<RVS OFF>-VASEMM
  ALLE   <RVS ON>/<RVS OFF>-OIKEALLE":REM EB
1126 PRINT"<12RIGHT>TAI JOYSTICK #2":REM 9A
1127 PRINT"<HOME><3DOWN><16RIGHT><RVS ON>F1<RVS OF
  F>-ALOITUS":REM 1B
1128 FORDE=0TOIM:REM E2
1130 I=I+1:IFI>9THENI=1:REM 5F
1140 PRINT"<HOME><RIGHT>"MID$(TX$,I,38):REM F2
1150 J=10-I:PRINT"<HOME><24DOWN><RIGHT>"MID$(TX$,J
  ,38)"<HOME>":REM 7B
```

```
1160 POKES,L%(0,DE):POKES+7,L%(1,DE):POKES+14,L%(2
  ,DE):REM 30
1165 POKES+1,H%(0,DE):POKES+8,H%(1,DE):POKES+15,H%
  (2,DE):REM 87
1170 POKES+4,C%(0,DE):POKES+11,C%(1,DE):POKES+18,C
  %(2,DE):REM A4
1175 GETA$:IFA$="<F1>"THEN2000:REM 38
1180 FORTT=1TO5:NEXT:NEXT:GOTO1128:REM DO
2000 POKEV+21,1:POKEV,165:POKEV+1,0:PRINT"<CLR/HOM
  E>";:REM 09
2002 FORI=STOS+24:POKEI,0:NEXT:REM 92
2005 POKEV+39,2:REM 41
2010 FORI=2TO14STEP2:POKEI+V,0:NEXT:REM E0
2020 POKE53280,3:POKE53281,15:REM 52
2025 GOSUB20000:REM 3B
2030 FORI=1TO24:PRINT"<LT.BLUE><RVS ON><9SPACE><RV
  S OFF><21RIGHT><RVS ON>▌<9SPACE>";:NEXT:REM 7D
2040 PRINT"<LT.BLUE><RVS ON><9SPACE><RVS OFF><21RI
  GHT><RVS ON>▌<8SPACE><HOME>":REM FC
2050 POKE2023,160:POKE2023+S,14:REM BF
2060 POKES+5,1:POKES+6,248:POKES+4,16:POKES+24,15:
  REM 7B
2070 PRINT"<HOME><8DOWN><18RIGHT><BLK> ╭─╮<DOWN><3L
  EFT> │● │<DOWN><3LEFT> │● │<DOWN><3LEFT> │● │<DOWN><3L
  EFT> ╰─╯":REM 27
2075 X1=100:X2=100:REM B7
2080 POKE55675,2:GOSUB10000:POKE55675+40,2:GOSUB10
  000:POKE55675+80,5:REM 84
2090 X1=100:X2=150:GOSUB10000:REM 64
2095 POKE53281,6:REM 64
2100 POKES,5:POKES+1,5:POKES+5,0:POKES+6,240:POKES
  +4,33:REM 23
2110 POKE5*4096,0:POKE6*4096,0:POKE16473,10:POKE49
  664,0:REM 46
2115 PRINT"<HOME><8DOWN><18RIGHT>    <DOWN><3LEFT>
     <DOWN><3LEFT>    <DOWN><3LEFT>    <DOWN><3LEFT>
     ":REM DF
2120 POKEV+21,255:SYS16384:SC=PEEK(5*4096)/2.5:IFS
  C>HSTHENHS=SC:REM 2D
2122 FORI=STOS+24:POKEI,0:NEXT:POKES+24,15:REM 74
```

```
2123 PRINT"<CLR/HOME>":REM 0A
2125 POKEV+21,0:POKE198,0:POKES+4,128:REM 82
2126 POKES,50:POKES+1,3:POKES+5,13:POKES+6,0:POKES
  +4,129:REM 5D
2130 FORI=1TO100:POKE53280,0:POKE53281,0:POKE53280
  ,1:POKE53281,1:NEXT:REM 31
2145 POKE53280,0:POKE53281,0:REM 21
2150 PRINT"<CLR/HOME><10DOWN><10RIGHT><ORANGE>PAAS
  IT"SC"KILOMETRIA.":REM 34
2160 PRINT"<3DOWN><8RIGHT>ENNATYS ON"HS"KILOMETRIA
  .":REM 72
2200 PRINT"<HOME><19DOWN><9RIGHT>PAINA JOTAIN NAPP
  AINTA":REM BA
2210 FORI=1TO500:NEXT:REM 14
2220 PRINT"<HOME><19DOWN><9RIGHT><RVS ON>PAINA JOT
  AIN NAPPAINTA<RVS OFF>":REM 60
2225 FORI=1TO500:NEXT:REM 1A
2230 GETA$:IFA$=""THEN2200:REM AE
2240 POKES+24,15:GOTO100:REM 71
10000 POKES,X1:POKES+1,X2:POKES+4,17:FORVI=1TO1000
  :NEXT:POKES+4,16:REM 5E
10010 FORVI=1TO1000:NEXT:RETURN:REM DD
20000 POKES,4:POKES+1,4:POKES+5,0:POKES+6,240:POKE
  S+4,33:POKES+24,15:REM 2F
20010 FORX=255TO200STEP-1:FORVI=1TO50:NEXT:POKEV+1
  ,X:NEXT:REM A1
20020 POKES+4,32:RETURN:REM 80
30000 PRINT"<CLR/HOME><12DOWN><12RIGHT>ODOTA HETKI
   !!!":POKE2040,15:RETURN:REM A9
58000 DATA,35,70,105,140,175,210:REM 49
59000 DATA34334,36376,38539,40830,43258,45830,4855
  6,51443,54502,57743,61176:REM 0B
59001 DATA64814:REM 20
59999 REM *** MUSIIKKIDATAT ***:REM CD
60000 DATA338,338,338,338,338,338,338,340,854,340,
  1106,596,596,340,340,340,340:REM 5D
60001 DATA596,601,1113,850,338,338,338,338,340,854
  ,340,1106,601,345,345:REM F5
60002 DATA345,345,345,343,598,596,1106,:REM C6
```

```
60005 DATA569,578,569,578,569,578,569,578,564,569,
  564,569,564,569,564,569:REM D5
60006 DATA569,578,569,578,569,578,585,594,1106,109
  9,585,583,1094,:REM 19
60010 DATA562,569,562,569,562,569,562,569,553,564,
  553,564,553,564,553,564:REM 99
60011 DATA562,569,562,569,562,569,578,585,1094,109
  2,585,569,1090,:REM 01
60999 REM *** KONEKIELIDATAT ***:REM FA
61000 DATA165,203,201,12,240,31,201,55,240,38,32,9
  5,64,32,134,64,173:REM 2F
61001 DATA0,220,201,123,240,14,201,119,240,21,32,9
  5,64,76,80,64,142:REM F4
61002 DATA3,64,96,174,0,208,202,142,0,208,76,42,64
  ,0,174,0,208:REM 0D
61003 DATA232,142,0,208,76,60,64,173,0,208,201,96,
  208,31,169,97,141:REM 14
61004 DATA0,208,76,80,64,234,234,0,173,0,208,201,2
  40,208,13,169,239:REM 0A
61005 DATA141,0,208,76,80,64,255,255,255,255,255,1
  62,0,160,0,232,200:REM 3A
61006 DATA208,253,224,2,208,248,76,2,65,96,197,201
  ,1,208,1,0,76:REM 53
61007 DATA134,64,0,0,234,0,234,160,3,190,0,208,208
  ,3,32,160,64:REM FD
61008 DATA232,138,153,0,208,200,200,192,16,48,237,
  76,240,63,174,0,194:REM 74
61009 DATA232,142,0,194,208,3,76,240,64,96,232,189
  ,232,64,141,21,208:REM 4C
61010 DATA142,0,194,76,166,64,142,0,192,140,1,192,
  172,0,193,200,192:REM 05
61011 DATA40,208,2,160,0,190,192,64,140,0,193,172,
  1,192,138,153,255:REM 04
61012 DATA207,174,0,192,96,192,112,80,158,112,96,2
  55,102,221,239,95,224:REM E5
61013 DATA170,102,144,255,143,223,79,191,111,159,1
  75,127,111,191,239,160,96:REM AA
61014 DATA224,176,192,176,208,128,112,160,240,96,3
  ,7,15,31,63,127,255:REM 7E
```

```
61015 DATA0,174,89,64,238,0,80,224,2,208,1,96,202,
  142,89,64,96:REM 2C
61016 DATA234,234,173,30,208,208,3,76,110,64,174,0
  ,96,232,224,32,48:REM 13
61017 DATA1,96,142,0,96,76,110,64,-1:REM 16
61999 REM *** SPRITEDATAT ***:REM 37
62000 DATA5,85,80,6,170,144,5,85,80,0,195,0,60,195
  ,60,60,170:REM B5
62001 DATA60,62,170,188,60,20,60,60,20,60,0,170,0,
  0,190,0,2:REM 5A
62002 DATA190,128,2,235,128,0,190,0,0,40,0,15,20,2
  40,15,40,240:REM EC
62003 DATA15,85,240,15,40,240,15,40,240,1,85,64,0,
  15,255,240,5:REM 03
62004 DATA85,80,15,255,240,0,195,0,60,195,60,60,17
  0,60,63,235,252:REM AD
62005 DATA60,20,60,60,20,60,0,170,0,0,190,0,1,195,
  64,1,125:REM 28
62006 DATA64,0,170,0,0,60,0,15,40,240,15,40,240,15
  ,255,240,15:REM BE
62007 DATA60,240,15,60,240,2,102,64,0,0,127,244,0,
  222,220,3,94:REM F4
62008 DATA215,3,122,183,3,122,183,3,122,183,3,234,
  175,3,234,175,3:REM A3
62009 DATA255,255,3,192,15,3,255,255,3,85,87,3,170
  ,171,3,170,171:REM 8A
62010 DATA3,85,87,3,255,255,3,192,15,3,255,255,3,1
  70,171,2,250:REM 1A
62011 DATA190,0,143,200,0,-1:REM 72
62500 ::REM 37
62501 ::REM 38
62502 ::REM 39
63000 REM ********************************:REM 9F
63001 REM *                              *:REM 32
63002 REM * (C) 1985 BY VESA TAKKINEN *:REM BD
63003 REM *                              *:REM 34
63004 REM ********************************:REM A3
```

Avaruuden valloitus

Pertti Kasanen

Olet saanut tehtäväksesi tutkia avaruuden kolkan, josta kukaan ei ole palannut. Vihamielisen sivilisaation itsemurhajoukot tukkivat tiesi. Tarkoituksenasi on tunkeutua mahdollisimman syvälle tuntemattomalle alueelle kartoittaaksesi tämän tähtikartalla alevan häpeällisen valkoisen laikun.

Peli on täysin konekielinen. Pelaajia voi olla yksi tai kaksi ja ohjaus tapahtuu kakkosportiin kytketyllä joystickillä. Pelaajamäärän valinta tapahtuu funktionäppäimillä ja pelin saa tilapäisesti pysäytettyä RUN/STOP-näppäimestä. Pelin alkuun pääseen painamalla RUN/STOP- ja RESTORE-näppäimiä yhtä aikaa.

Selviytyäksesi hyökkäysaallosta sinun on joko ammuttava kaikki vihollisalukset tai pujahdettava vihollisen rivistöjen lävitse. Mitä pidemmälle pääset, sitä raivokkaammiksi ja nopeammiksi vihollisalukset tulevat. Yli kymmennen tuhannen pisteen tulosta on jo pidettävä erinomaisena.

Ohjelmassa on käytetty lähes kaikkia Kuusnelosen tarjoamia grafiikkaominaisuuksia: tavallista ja multicoloria niin merkeissä kuin spiteissäkin, törmäyksiä, rasterikeskeytystä, smooth-scrollausta, muutettu tekstin taustaväriä, muutettu keskeytykset liikuttamaan grafiikkaa jne. Monet käytetyistä tempuista ovat mahdollisia vain konekielellä.

Kuva pelin tulostuksesta on myös tämän kirjan kannessa oikealla puolella.

```
PROGRAM: AVARUUDEN VALL.

0801: 0B08C1079E323036 = 211
0809: 310000004C3A0930 = 0F0
0811: 3030303030000000 = 0F0
0819: 0000000000000000 = 000
0821: 0000001E0864B4B4 = 1F2
0829: DC00000000050505 = 0EB
0831: 0000000000000000 = 000
0839: 000000000000AD35 = 0E2
0841: 084902204112AD00 = 173
0849: DC490F482901F00D = 2A3
0851: AD01D0C9669006CE = 411
0859: 01D0CE03D0684829 = 34B
0861: 02F00DAD01D0C9D7 = 41D
0869: B006EE01D0EE03D0 = 436
0871: AD10D02903D02C68 = 31D
0879: 482904F00DAD00D0 = 2EF
0881: C9289006CE00D0CE = 3F3
0889: 02D0682908F013EE = 35C
0891: 00D0EE02D0AD00D0 = 40D
0899: D008A9030D10D08D = 2FE
```

```
08A1: 10D06068482904F0 = 30D
08A9: 15CE00D0CE02D0AD = 400
08B1: 00D0C9FFD008A9FC = 515
08B9: 2D10D08D10D06829 = 30B
08C1: 08F0DFAD00D0C929 = 446
08C9: B0D8EE00D0EE02D0 = 506
08D1: 60A5C5C93FF00160 = 423
08D9: A9018D3708A5C5C9 = 3A9
08E1: 3FF0FAA5C5C93FD0 = 56B
08E9: FAA5C5C93FF0FAA9 = 5FF
08F1: 008D370860A9018D = 263
08F9: 380820D208204010 = 1AA
0901: 203F08208F0F205B = 1A0
0909: 0920D00C20C20920 = 210
0911: 321220AB11AD2E08 = 203
0919: D014AD3D08F00BAD = 37E
0921: 2F08186D3008C902 = 1BF
0929: B00420131360AD36 = 23D
0931: 08F0C22032134CFB = 366
0939: 08207B1520F20B20 = 1F5
0941: DC1220320A209812 = 214
0949: 20DC1220281520C0 = 24B
0951: 0D20F60820CE144C = 279
0959: 4309AD32080A0A0A = 151
0961: 1869258562AD3408 = 276
0969: 2907186562CD05D0 = 2B1
0971: 9048AD00DC2910D0 = 36A
0979: 38AD03D038E9088D = 36E
0981: 05D0AD10D0A82903 = 336
0989: F005980904D00398 = 305
0991: 29FB8D10D0AD02D0 = 410
0999: 1869078D04D09008 = 281
09A1: AD10D009048D10D0 = 307
09A9: AD35080904204112 = 16A
09B1: 60AD350829FB2041 = 2CF
09B9: 1260CE05D0CE05D0 = 3B8
09C1: 60AD2D08186D2808 = 1F7
09C9: 8D2D08900DAC07D0 = 2E2
09D1: C8C0EC9002A0488C = 47A
09D9: 07D0AD1608D02CAD = 34B
09E1: 32080A0A0A186929 = 102
09E9: 186D2A088562AD34 = 27F
09F1: 0829071865628562 = 1FE
09F9: AD07D038E562900A = 39D
0A01: C903B006AD03D08D = 38F
0A09: 160860CD07D0D0FA = 3EC
0A11: 8D0BD0AD10D029DF = 3FD
0A19: 8D10D0A91A8D0AD0 = 397
0A21: AD35080920204112 = 186
0A29: A9008D1608209014 = 218
0A31: 60A93A8D0203A909 = 287
0A39: 8D0303A9188D16D0 = 2C7
0A41: AD11D029708D11D0 = 395
0A49: A9058D2E088D2F08 = 235
0A51: 8D3008A9A08D00D0 = 36B
0A59: 8D02D0A9C88D01D0 = 42E
0A61: 8D03D0A91E8D07D0 = 38B
0A69: A9138D06D0A9008D = 355
0A71: 08D08D04D08D05D0 = 39B
0A79: A9E78D09D0A9058D = 431
0A81: 27D0A9088D28D0A9 = 3D6
0A89: 0F8D29D0A90D8D2A = 302
0A91: D08D2BD0A90E8D2C = 3C8
0A99: D08D2DD0A90F8D18 = 3B7
0AA1: D4A9008D10D08D1C = 393
0AA9: 088D1E088D20088D = 1FD
0AB1: 1D088D1F088D2108 = 18F
0AB9: 8D18088D16088D1A = 1FF
0AC1: 088D3A088D3B088D = 234
0AC9: 3E088D36088D3708 = 1DD
0AD1: 8D38088D20D08D21 = 2F8
0AD9: D0A9068D22D0A90E = 3B5
0AE1: 8D23D0A9038D1CD0 = 3A5
0AE9: A9078D25D0A9088D = 370
0AF1: 26D0AD24088D2208 = 286
0AF9: AD25088D2308A901 = 23C
0B01: 8D3C08A20FA90420 = 24F
0B09: D613A03FA90099C0 = 3CA
0B11: DB88D0FAA007A2C8 = 53E
0B19: 8A99F807CA8810F8 = 47C
```

```
0B21: A91B8D15D08D3508 = 300
0B29: AD18D029F0090C8D = 350
0B31: 18D0AD1ED0AD1FD0 = 41F
0B39: 60AD3708F00160AD = 34A
0B41: 35082920F030AD10 = 263
0B49: D0A82920D013AD0A = 35B
0B51: D018690A8D0AD090 = 352
0B59: 1D9809208D10D0D0 = 31B
0B61: 15AD0AD018690A8D = 2B4
0B69: 0AD0C9799008AD35 = 396
0B71: 0829DF204112AD35 = 265
0B79: 082940F01DAD0DD0 = 308
0B81: 38E9068D0DD0C92E = 388
0B89: B010AD350829BF20 = 2B2
0B91: 4112AD10D029BF8D = 355
0B99: 10D060AD2B08D007 = 2F7
0BA1: AD12D0C9FCD024A9 = 4F1
0BA9: 1F8D18D0AD340809 = 286
0BB1: 478D11D0A9008D15 = 300
0BB9: D0A95E8D12D0A901 = 3F0
0BC1: 8D19D0A9088D16D0 = 39A
0BC9: 4C31EAA91C8D18D0 = 3A1
0BD1: AD34088D11D0AD35 = 339
0BD9: 088D15D0A9FC8D12 = 3BE
0BE1: D0A9018D19D0A918 = 3B1
0BE9: 8D16D0203A0B4C31 = 255
0BF1: EA78A99C8D1403A9 = 3F4
0BF9: 0B8D1503A9018D1A = 201
0C01: D0A9FC8D12D0AD11 = 4A2
0C09: D029778D11D08D34 = 39F
0C11: 08AD0EDC29FE8D0E = 361
0C19: DC5860AD2C08A000 = 315
0C21: 18698008186D2708 = 1BD
0C29: 8D2C089001C82890 = 2D2
0C31: 01C8AD10D02910D0 = 35F
0C39: 1298186D08D0A890 = 33F
0C41: 21AD10D009108D10 = 264
0C49: D0D01798186D08D0 = 3AC
0C51: A8C079900DA0008C = 3AA
0C59: 1A08AD10D029EF8D = 354
0C61: 10D08C08D060AD1A = 36B
0C69: 08D063AC10D09829 = 388
0C71: 03D022982910D00D = 2A3
0C79: AD02D038ED08D0B0 = 42C
0C81: 31A9FFD02DA97938 = 430
0C89: ED08D0186D02D090 = 3AC
0C91: 21A9FFD01D982910 = 387
0C99: F00DAD02D038ED08 = 3A9
0CA1: D0B00FA9FFD00BAD = 4BF
0CA9: 02D038ED08D09002 = 361
0CB1: A9FF38ED2608B016 = 3C1
0CB9: A000AD02D029FE8D = 3D3
0CC1: 1808AD10D02903F0 = 2C9
0CC9: 02A0108C19086020 = 1DF
0CD1: 1C0CAD1808A80D19 = 1C3
0CD9: 08D0034C670C98CD = 2FF
0CE1: 08D0D03DAD10D029 = 39B
0CE9: 10CD1908D0330A0A = 215
0CF1: AAAD10D029BF8D10 = 3BC
0CF9: D08A0D10D08D10D0 = 3B4
0D01: 988D0CD0A9E08D0D = 424
0D09: D0AD350809402041 = 264
0D11: 12A9018D1A08A900 = 214
0D19: 8D18088D19082090 = 20B
0D21: 1460A20F8669CAD0 = 3AE
0D29: FDC669D0F960AD34 = 536
0D31: 08186901C918D002 = 23D
0D39: A9108D340860A006 = 288
0D41: 844E203F0820230D = 189
0D49: 203F08202F0DAD1F = 18F
0D51: D02901D00DAD1ED0 = 372
0D59: AA2901F0328A2960 = 309
0D61: F02D6868204211AD = 30D
0D69: 3D08F00EAD2F0818 = 23F
0D71: 6D3008C902900E4C = 25A
0D79: 3213AD2E08F00620 = 23E
0D81: FD134C0C0EAD3808 = 263
0D89: F00268684C1313C6 = 2FA
0D91: 4ED0AF20230D6020 = 29D
0D99: 3F08202F0D203F08 = 10A
```

```
0EE1: 856360AD34082907 = 261   0DA1: AD34082907D0F0A9 = 382
0EE9: C907F004202F0D60 = 280   0DA9: C88565A9F08567A9 = 4E0
0EF1: 202F0D20BE0EA8A5 = 295   0DB1: 0485668568A0F0B1 = 41D
0EF9: 6218692885649001 = 285   0DB9: 65916788D0F960AD = 4BB
0F01: C88465A562853EA5 = 420   0DC1: 3A08F0238531A924 = 2D8
0F09: 631869D4853F2076 = 312   0DC9: 38ED31088532A005 = 2BA
0F11: 0E20760E20760E20 = 176   0DD1: 205012F8A53138E9 = 371
0F19: 760EEE320860CE32 = 30C   0DD9: 018531D8D0F0AD3A = 436
0F21: 08CE3208A56238E9 = 338   0DE1: 088531C632D0E7F8 = 465
0F29: 288562A023A90491 = 310   0DE9: AD3A08186901BD3A = 238
0F31: 628810FBA9108566 = 399   0DF1: 08D8A201BD660418 = 2C2
0F39: 20E40E203F082023 = 1BC   0DF9: 6901C93AD00AA930 = 320
0F41: 0D203F08203515C6 = 1A4   0E01: 9D6604CA10EE3003 = 302
0F49: 66D0ED60A9008D39 = 3F2   0E09: 9D6604A9008D08D0 = 315
0F51: 08AD350829FB2041 = 277   0E11: 8D1A088D18088D19 = 202
0F59: 1220E40E203F0820 = 1AB   0E19: 088D1608AD10D029 = 269
0F61: 230D203F08203515 = 101   0E21: EF8D10D0A9488D07 = 3E1
0F69: AD3208C91090DDA9 = 3D6   0E29: D020E314A904854F = 368
0F71: 018D3908201F0F20 = 13D   0E31: 8563A9C985622098 = 3F9
0F79: 1F0F201F0FA56238 = 1BB   0E39: 0DA024A904889162 = 2F9
0F81: E9288562A023A904 = 368   0E41: D0FB203F0D20980D = 2FC
0F89: 91628810FB60A900 = 38F   0E49: A023A904916288A9 = 394
0F91: 852B20BE0EA023B1 = 310   0E51: 01916288A9009162 = 318
0F99: 62C904D01488D0F7 = 462   0E59: 88A90491628810EC = 3AC
0FA1: E62BE62BA5621869 = 3AA   0E61: 203F0DC64FD0CFA9 = 3C9
0FA9: 50856290E8E663D0 = 4C8   0E69: 058D320820E40EA9 = 287
0FB1: E4AD320818652B0A = 27D   0E71: 248D310860A023B1 = 2BE
0FB9: 0A0A691F85628562 = 26A   0E79: 629164A9049162B1 = 3A8
0FC1: AD34082907186562 = 1F8   0E81: 3EC90FF012AA9818 = 372
0FC9: CD03D09006204D0F = 2B2   0E89: 6928A88A913E9838 = 362
0FD1: 20C00D60C9049015 = 2BF   0E91: E928A8A90F913E88 = 3C8
0FD9: D00160A5631869D4 = 38E   0E99: 10DDA56218695090 = 355
0FE1: 853DA562853CA90F = 342   0EA1: 02E6638562A46318 = 351
0FE9: 913CA90460490360 = 286   0EA9: 692885649001C884 = 357
0FF1: A023B16220D50FAA = 384   0EB1: 65A53E186950853E = 2DC
0FF9: 88B16220D50FC891 = 3F8   0EB9: 9002E63F60A90085 = 345
1001: 6288D0F48A9162A0 = 4CB   0EC1: 63AD32080A0A0A85 = 1ED
1009: 50B16220D50FAAC8 = 3D9   0EC9: 620A26630A266318 = 1A0
1011: B16220D50F889162 = 392   0ED1: 65629002E6631869 = 323
1019: C8C8C07490F2888A = 558   0ED9: 028562A563186904 = 276
```

```
1021: 91626020BE0E20F1 = 350  1161: 04A9C88D01D08D03 = 363
1029: 0FAD22082901F00E = 20E  1169: D0A9A08D00D08D02 = 405
1031: A5621869A09002E6 = 3A0  1171: D0AD10D029FC8D10 = 41F
1039: 63856220F10F60CE = 398  1179: D0AD3508291C2041 = 260
1041: 2308D00AAD25088D = 26C  1181: 12A9488D07D0A900 = 310
1049: 230820241060CE22 = 1CF  1189: 8D08D08D18088D19 = 2B8
1051: 08D009AD24088D22 = 269  1191: 088D16088D1A08AD = 20F
1059: 0820E40E60AD3508 = 264  1199: 10D029EF8D10D0AD = 412
1061: 29FB204112A9008D = 2CD  11A1: 3D08F005A9018D36 = 2A7
1069: 05D060A562853CA5 = 3A2  11A9: 0860AD1FD085FC29 = 3AE
1071: 631869D4853DA001 = 31B  11B1: 04F039AD05D0F034 = 3D3
1079: B162C904B032C901 = 38C  11B9: 20CD10AD3308F02C = 301
1081: F004C902D015A907 = 354  11C1: 205E1020AF14AD1C = 23A
1089: 9162A901913C88A9 = 39B  11C9: 0818690A8D1C0890 = 1D4
1091: 059162A901913CEE = 35D  11D1: 03EE1D08A0042050 = 22A
1099: 330860A9059162A9 = 2E5  11D9: 12CE3108D00EAD22 = 2C6
10A1: 01913CC8A9079162 = 339  11E1: 0809018D22082024 = 10D
10A9: A901913CEE330860 = 300  11E9: 1020C00DA5FC2901 = 2C8
10B1: C8B162C904B014A9 = 415  11F1: F013204211AD3D08 = 268
10B9: 059162A901913CC8 = 337  11F9: D00BAD2E08F00620 = 2D4
10C1: A9079162A901913C = 31A  1201: FD134C0C0EAD1ED0 = 311
10C9: EE330860AD340829 = 29B  1209: 85FD2904F003205E = 320
10D1: 078564A9008563AD = 32E  1211: 10A90125FDF019A5 = 38A
10D9: 05D038E92F38E564 = 3A6  1219: FD2960F013204211 = 2FC
10E1: 29F885620A26630A = 2A5  1221: AD3D08D00BAD2E08 = 2B0
10E9: 2663186562856290 = 2DF  1229: F00620FD134C0C0E = 28C
10F1: 02E663A563186904 = 2D8  1231: 60A20B8669AE2908 = 2DB
10F9: 8563AD04D038E918 = 3A2  1239: CAD0FDC669D0F660 = 5EC
1101: 084A4A4A65628562 = 294  1241: 8D3508AD15D0F006 = 352
1109: 9002E663289012AD = 352  1249: AD35088D15D060A2 = 35E
1111: 10D02904F00BA562 = 30F  1251: 05AD3E08F002A222 = 2AE
1119: 18692085629002E6 = 300  1259: 862DA904852E8530 = 2C8
1121: 63A462D002C66388 = 3EC  1261: A914852FB12D1869 = 2D0
1129: 8462A9008D330820 = 277  1269: 01C97AD009A97091 = 3C7
1131: 6C10A56218692885 = 2B1  1271: 2D8810F03002912D = 2A5
1139: 629002E663206C10 = 2D9  1279: A000B12D29BFD12F = 366
1141: 60CE2E08A9008D05 = 29F  1281: 9014D007C8C006F0 = 3F9
1149: D0AD350829FB2041 = 33F  1289: 0DD0EFB12D29BF91 = 423
1151: 1220FF15AD3E08F0 = 329  1291: 2FC8C006D0F560A2 = 484
1159: 05CE7604D003CE58 = 346  1299: 00A92020D613A901 = 27C
```

```
12A1: 8D2B08AD16D02910 = 28C
12A9: D0F9A20DA92020D6 = 437
12B1: 13A00DB9DF2F99EC = 40C
12B9: 0588D0F7A00EB9EF = 4AA
12C1: 2F993C0688D0F7A5 = 3FE
12C9: C5C904D004A900F0 = 3FF
12D1: 06C905D0F2A9018D = 3CD
12D9: 3D0860A90385FDA9 = 37C
12E1: D785FFA9FF85FC85 = 609
12E9: FEA0C8A90D91FEB9 = 564
12F1: FF2E91FC88D0F4A9 = 5AF
12F9: 9F85FEE6FFA028A9 = 578
1301: 0091FE88D0F9A006 = 486
1309: B90F0899130488D0 = 2D8
1311: F760A006B9130499 = 366
1319: 0F0888D0F760AD00 = 373
1321: DC291F491FD0F7AD = 400
1329: 00DC291F491FF0F7 = 373
1331: 60A9008D3608AD3E = 2BF
1339: 08D04DA9018D3E08 = 2A2
1341: AD1C088D1E08AD2E = 25F
1349: 088D2F08AD3A088D = 248
1351: 3B08AD20088D1C08 = 1C9
1359: AD30088D2E08F8AD = 34D
1361: 3C088D3A08290FD8 = 223
1369: 1869308D6704F8AD = 34E
1371: 3C0829F0D84A4A4A = 313
1379: 4A1869308D660420 = 212
1381: 551420FD134C0C0E = 1FF
1389: A9008D3E08AD1C08 = 24D
1391: 8D2008AD2E088D30 = 255
1399: 08AD3A088D3C08AD = 275
13A1: 1E088D1C08AD2F08 = 1BB
13A9: 8D2E08F8AD3B088D = 338
13B1: 3A08290FD8186930 = 203
13B9: 8D6704F8AD3B0829 = 309
13C1: F0D84A4A4A4A1869 = 371
13C9: 308D660420551420 = 1D0
13D1: FD134C0C0E8537A9 = 2DB
13D9: 048534A9D88536A9 = 3A2
13E1: C885338535A0008A = 364
13E9: 9135A53791338BD0 = 3BE
13F1: F6E636E634A534C9 = 4CE
13F9: 07D0EC60A200A920 = 38E
1401: 20D613A9018D2B08 = 273
1409: AD16D02910D0F9A2 = 437
1411: 0DA92020D613A009 = 288
1419: B9C72F99EE0588D0 = 493
1421: F7AD3E08F005A932 = 3BA
1429: 8DF705201F13A200 = 27D
1431: A90420D613A9008D = 2EC
1439: 2B08AD16D02910F0 = 2EF
1441: F9A20FA90420D613 = 360
1449: AD35080903204112 = 169
1451: AD1FD060A000AD3E = 387
1459: 08F002A01C843898 = 30A
1461: 491C853AA9048539 = 28F
1469: 853BA204A00BB138 = 2FA
1471: 09409138B13A29BF = 2E5
1479: 913A8810F1A53818 = 349
1481: 69288538A53A1869 = 2AE
1489: 28853ACAD0DE60A9 = 468
1491: 008D04D4A90F8D01 = 2AB
1499: D4A9D48D00D4A97A = 4D5
14A1: 8D05D4A9008D06D4 = 376
14A9: A9818D04D460A900 = 398
14B1: 8D0BD4A9078D08D4 = 385
14B9: A9008D07D4A9598D = 3A0
14C1: 0CD4A9008D0DD4A9 = 3A0
14C9: 818D0BD460A201A9 = 399
14D1: 0420D613A0A0B9FF = 405
14D9: 0329BF99FF0388D0 = 3DE
14E1: F560AD3A08290F85 = 301
14E9: FFAD3A084A4A4A4A = 316
14F1: A8F008A90018690A = 2D4
14F9: 88D0FA1865FFC91F = 4B6
1501: 9002A91F0A85FEA9 = 390
1509: 6438E5FE8D260806 = 340
1511: FEA9701865FE8D27 = 446
1519: 088D280806FEA9FF = 371
```

```
1521: 38E5FE8D29086020 = 359   1661: 323A303030303030 = 18C
1529: 551420FD13AD1ED0 = 334   1669: 2020202020202020 = 100
1531: AD1FD060AD1FD029 = 3C1   1671: 202020201B202020 = 0FB
1539: 01D00DAD1ED0A829 = 34A   1679: 2020202020202020 = 100
1541: 01F036982960F031 = 369   1681: 2020201C20202020 = 0FC
1549: AD3908F002686868 = 318   1689: 2020202020202020 = 100
1551: 686868204211AD3D = 295   1691: 2008050E0B09013A = 08A
1559: 08F00EAD2F08186D = 26F   1699: 352020201B202020 = 110
1561: 3008C902900E4C32 = 21F   16A1: 1401130F3A203030 = 0F1
1569: 13AD2E08F00620FD = 309   16A9: 2020201C20202008 = 0E4
1571: 134C0C0E68684C13 = 1A8   16B1: 050E0B09013A3520 = 0B7
1579: 1360A99320D2FFA9 = 449   16B9: 1D1D1D1D1D1D1D1D = 0E8
1581: 008D20D08D21D0A2 = 39D   16C1: 1D1D1D1D1E1D1D1D = 0E9
1589: 01A92020D613A941 = 2BD   16C9: 1D1D1D1D1D1D1D1D = 0E8
1591: 85FCA91685FDA92F = 49A   16D1: 1D1D1D1F1D1D1D1D = 0EA
1599: 85FFA00084FEB1FC = 553   16D9: 1D1D1D1D1D1D1D1D = 0E8
15A1: 91FEC8D0F9E6FDE6 = 6E9   16E1: 0404040404040404 = 020
15A9: FFA5FFC932D0EFA0 = 5FD   16E9: 0404040404040404 = 020
15B1: 30B1FC91FE8810F9 = 4FD   16F1: 0404040404040404 = 020
15B9: A5FC18693085FC90 = 463   16F9: 0404040404040404 = 020
15C1: 02E6FDA93885FFA0 = 4EA   1701: 0404040404040404 = 020
15C9: 00B1FC91FEC8D0F9 = 5CD   1709: 10050C01010A0120 = 04E
15D1: E6FDE6FFA5FFC93A = 66F   1711: 3120202020202020 = 111
15D9: D0EFAD18D029F009 = 476   1719: 2020202020202020 = 100
15E1: 0F8D18D0A010B900 = 2ED   1721: 0631292031201005 = 0E6
15E9: 32999C058810F7A0 = 39B   1729: 0C01010A01202020 = 079
15F1: 12B91A3299EB0588 = 328   1731: 0633292032201005 = 0E9
15F9: 10F7201F1360AD35 = 29B   1739: 0C01010A01012020 = 05A
1601: 0829FD204112A900 = 24A   1741: 828A2B0F8DAF030C = 291
1609: 8D0BD48D0DD48D07 = 36E   1749: 0888A0C0C8E800C0 = 460
1611: D4A9098D08D4A97A = 412   1751: 0080A8C8C0E808C0 = 460
1619: 8D0CD4A9818D0BD4 = 403   1759: 020AAB8F0DAF830C = 291
1621: A9C0854DA200A000 = 37D   1761: 0000000000000000 = 000
1629: EE25D0EE26D0EE27 = 4DC   1769: 301C0F0F1F7BC303 = 1CA
1631: D088D0F4CAC64DD0 = 5C9   1771: 0000000000000000 = 000
1639: EBAD1ED0AD1FD060 = 482   1779: 1870F0E0F0BC0C00 = 410
1641: 10050C313A303030 = 11C   1781: 0010000054000054 = 0B8
1649: 303030201B201001 = 0FC   1789: 0000540001750001 = 0CB
1651: 1201133A30303030 = 120   1791: 750001FD0009FD80 = 2F9
1659: 3030201C2010050C = 0DD   1799: 29FDA0A9FDA8A9FD = 5BA
```

```
17A1: A8A955A800A80000 = 2F6
17A9: 0000000000000000 = 000
17B1: 0000000000000000 = 000
17B9: 0000000000000000 = 000
17C1: 0000000000000000 = 000
17C9: 0000000000000000 = 000
17D1: 0000000000000000 = 000
17D9: 0000000000000000 = 000
17E1: 0000000000000000 = 000
17E9: A80000A800006400 = 1B4
17F1: 0020000020000010 = 050
17F9: 0000000000000000 = 000
1801: 1000002800003800 = 070
1809: 0010000010000010 = 030
1811: 0000380000440000 = 07C
1819: 0000000000000000 = 000
1821: 0000000000000000 = 000
1829: 0000000000000000 = 000
1831: 0000000000000000 = 000
1839: 0000000000000000 = 000
1841: 00000000000003E0 = 0E3
1849: 0003F00000F80001 = 1EC
1851: FC0001F80001F800 = 2EE
1859: 00FC0000FE0000FF = 2F9
1861: 0000FE0000FC0001 = 1FB
1869: F80001F80001FC00 = 2EE
1871: 00F80003F00003E0 = 2CE
1879: 0000000000000000 = 000
1881: 0000000000000000 = 000
1889: 0000000000000000 = 000
1891: 1000003800027C80 = 146
1899: 07FFC00FFFE01FFF = 4D2
18A1: F01FFFF01CC6701C = 46C
18A9: 0070000000000000 = 070
18B1: 0000000000000000 = 000
18B9: 0000000000000000 = 000
18C1: 0000000000000000 = 000
18C9: 0000000000000000 = 000
18D1: 0008000042000210 = 05C
18D9: 00008400000EEFBF = 240
18E1: FF00000E00012400 = 132
18E9: 0008000021000008 = 031
18F1: 0000000000000000 = 000
18F9: 0000000000000000 = 000
1901: 005900013C00007E = 114
1909: 80013C00001A0000 = 0D7
1911: 5880001800009A00 = 18A
1919: 0018000018000018 = 048
1921: 0000000000180000 = 018
1929: 1800001800001800 = 048
1931: 0018000000000018 = 030
1939: 0000180000180000 = 030
1941: 0116011215150405 = 05D
1949: 0E2016010C140115 = 07B
1951: 1320243345292C59 = 17D
1959: 002428032920102E = 0D6
1961: 0B0113010E050E2C = 06D
1969: 2031393835203400 = 14B
1971: FEC2CAEEE0E6FE00 = 63C
1979: 7E6666FEE6E6E600 = 4FA
1981: FCCCCCFEE6E6FE00 = 65C
1989: FEC6C0E0E0E6FE00 = 628
1991: FEC6C6E6E6E6FE00 = 63A
1999: 7E6060F8E0E0FE00 = 4F4
19A1: 7E6060F8E0E0E000 = 4D6
19A9: 7E6660EEE6E6FE00 = 4FC
19B1: 666666FEE6E6E600 = 4E2
19B9: 1818183838383800 = 128
19C1: 1818181C1CDC7800 = 1D4
19C9: CCCCCCFEE6E6E600 = 614
19D1: 606060E0E0E0FE00 = 4BE
19D9: 7F4949C9C9C9C900 = 435
19E1: 7E6666E6E6E6E600 = 4E2
19E9: 7E6666E6E6E6FE00 = 4FA
19F1: 7E6666FEE0E0E000 = 4E8
19F9: 7E6262E2E2EAFE0C = 4FA
1A01: FCCCCCFEE6E6E600 = 644
1A09: FEC6C0FE0ECEFE00 = 55C
1A11: FE30303838383800 = 23E
1A19: 666666E6E6E6FE00 = 4E2
```

```
1A21: 666666E6E6FC3800 = 432  1AD1: 0000000000181830 = 060
1A29: 494949C9C9C9FF00 = 435  1AD9: 0000007E00000000 = 07E
1A31: C6C6C638E6E6E600 = 53C  1AE1: 0000000000181800 = 030
1A39: C6C6C6FE38383800 = 3F8  1AE9: 0003060C18306000 = 0BD
1A41: FEC606FEE0E6FE00 = 58C  1AF1: 7E6666E6E6E6FE00 = 4FA
1A49: C0C0C0C0C0C0C0C0 = 600  1AF9: 1818183838383800 = 128
1A51: 0303030303030303 = 018  1B01: FEC606FEE0E0FE00 = 586
1A59: 000000000000FFFF = 1FE  1B09: FE06063E0E0EFE00 = 262
1A61: C0C0C0C0C0C0FFFF = 67E  1B11: C6C6C6FE0E0E0E00 = 37A
1A69: 030303030303FFFF = 210  1B19: FEC6C0FE0E0EFE00 = 49C
1A71: 0000000000000000 = 000  1B21: FEC6C0FEE6E6FE00 = 64C
1A79: 1818383838003800 = 110  1B29: FEC60C1838383800 = 290
1A81: 6666660000000000 = 132  1B31: 7C6464FEE6E6FE00 = 50C
1A89: 6666FF66FF666600 = 3FC  1B39: FEC6C6FE0ECEFE00 = 562
1A91: 187E407E067E1800 = 1F0  1B41: 0000180000180000 = 030
1A99: 62660C1830664600 = 1C8  1B49: 000018000018183O = 078
1AA1: 3C663C3867663F00 = 222  1B51: 0E18306030180E00 = 10C
1AA9: 060C180000000000 = 02A  1B59: 00007E007E000000 = 0FC
1AB1: 0C18303030180C00 = 0D8  1B61: 70180C060C187000 = 12E
1AB9: 30180C0C0C183000 = 0B4  1B69: FEC60C1838003800 = 258
1AC1: 00663CFF3C660000 = 243  1B71: 00C7C7C7C7C7C7C7 = 571
1AC9: 0018187E18180000 = 0DE
```

Pong

Jyrki J. J. Kasvi

Ohjelmassa pelaajan tehtävänä on estää laatikossa poukkoilevaa palloa karkaamasta siitä ulos. Käytettävänä on maila, jonka avulla pallo voidaan kimmauttaa takaisin. Valitettavasti pallo vain saa lisää vauhtia jokaisesta kimmahduksesta.

Ohjaus tapahtuu kakkosporttiin kytketyn joystickin avulla.

```
PROGRAM:PONG

1 GOTO1000:REM 2B
10 IF(PEEK(JS)AND1)=0ANDMY>58THENMY=MY-2:REM 00
20 IF(PEEK(JS)AND2)=0ANDMY<200THENMY=MY+2:REM 23
30 PX=PX+XS:PY=PY+YS:REM 66
50 POKEV+1,MY:POKEV+3,PY:IFPX<256THENPOKEV+2,PX:PO
  KEV+16,1:GOTO70:REM C1
60 POKEV+2,PX-256:POKEV+16,3:REM 96
70 IFPEEK(V+31)=2THEN200:REM 81
80 IFPEEK(V+30)=3THEN500:REM 85
90 IFPX<300THEN10:REM FF
100 PRINT"<HOME><10DOWN><15RIGHT>GAME OVER!":REM 4
  9
110 INPUT"<2DOWN><2RIGHT>HALUATKO UUDEN PELIN K/E"
  ;VA$:IFVA$<>"K"ANDVA$<>"E"THEN110:REM A8
120 RESTORE:REM B7
130 IFVA$="K"THEN1000:REM 9A
140 PRINT"<CLR/HOME>":END:REM 0A
200 XS=XS+.2:YS=YS+.2:IFPX<32THENXS=ABS(XS):GOTO10
  :REM D9
210 IFPY>221THENYS=-ABS(YS):GOTO10:REM 8A
220 YS=ABS(YS):GOTO10:REM 24
500 IFABS(PY+21-MY)>20THENYS=1.3*YS:XS=-ABS(XS):GO
  TO10:REM 6A
510 YS=YS/1.3:XS=-ABS(XS):GOTO10:REM E1
1000 PRINT"<CLR/HOME><PUR>
```

```
  ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒":REM 0B
1010 FORSI=1TO23:PRINT"▒":NEXT:REM E5
1020 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
  ";:REM 19
1040 V=53248:MY=150:MX=24:PX=100:PY=170:JS=56321:P
  OKEV+28,3:POKEV+37,7:REM 7A
1050 POKEV+21,3:POKEV+32,1:POKEV+33,1:POKEV+16,1:P
  OKEV+23,1:POKEV,MX:POKEV+38,0:REM 2C
1060 POKE2040,13:POKE2041,14:POKEV+39,2:POKEV+40,5
  :REM 70
1070 XS=RND(0)*3+2:YS=RND(0)*3+1:IFRND(0)>.5THENXS
  =-XS:REM CE
1080 IFRND(0)>.5THENYS=-YS:REM 4F
1100 FORSI=64*13TO15*64-2:READSD:POKESI,SD:NEXT: G
  OTO10:REM 78
10000 REM MAILA:REM 39
10010 DATA 170,128,0,147,128,0,147,128,0,147,128,0
  ,147,128,0,147:REM 3B
10020 DATA 128,0,147,128,0,147,128,0,147,128,0,147
  ,128,0,147,128:REM 3F
10030 DATA 0,147,128,0,147,128,0,147,128,0,147,128
  ,0,147,128,0:REM D5
10040 DATA 147,128,0,147,128,0,147,128,0,147,128,0
  ,170,128,0,0:REM D2
10050 REM PALLO:REM 52
10060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:REM A5
10070 DATA 0,0,0,40,0,0,170,0,2,170,128,10,245,160
  ,10,245:REM B6
10080 DATA 160,10,95,160,10,95,160,2,170,128,0,170
  ,0,0,40,0:REM 2B
10090 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:REM A8
```

Meteors

Jyrki J. J. Kasvi

Avaruusaluksesi on joutunut meteorimyrskyyn ja yksikin törmäys tietää varmaa tuhoa. Kaiken muun lisäksi meteorit vielä nopeutuvat sitä mukaa, mitä syvemmälle meteoriparveen joudutaan.

Aluksen ohjaus tapahtuu kursorinäppäinten avulla. Ylös/alas-näppäin siirtää alusta vasemmalle ja vasen/oikea-näppäin oikealle.

```
PROGRAM:METEORS

0 PRINT"<CLR/HOME>":GOTO1000:REM C8
10 FORSI=1TOVA:NEXT:VA=VA-.5:REM 3A
20 PRINTTAB(RND(0)*38)"";:ON INT(RND(0)*5+1)GOTO 3
  0,40,50,60,65:REM A9
30 PRINT"<WHT>♠":GOTO70:REM D4
40 PRINT"<RED>♥":GOTO70:REM FE
50 PRINT"<RED>♦":GOTO70:REM 06
60 PRINT"<WHT>♠":GOTO70:REM EE
65 PRINT:REM F8
70 GET SU$:REM 13
80 IF SU$="<RIGHT>" THEN X=X+4:REM DC
90 IF SU$="<DOWN>" THEN X=X-4:REM D3
100 IF X<256 THEN POKE V+16,0: POKE V,X: GOTO120:R
  EM 3C
110 POKE V+16,1: POKE V,X-256:REM 13
120 IF PEEK(V+31)<>0 THEN 200:REM E8
130 GOTO 10:REM 2E
200 FORSI=1TO75:POKEV+39,RND(0)*16:POKEV+37,RND(0)
  *16:POKEV+38,RND(0)*16:NEXT:REM 70
210 PRINT"<CLR/HOME><13DOWN><15RIGHT>GAME OVER!":R
  EM FE
215 KI=(70-VA)*2-22:IFKI<0THENKI=0:REM CB
220 PRINT"<2DOWN>SAIT VAISTETTYA"KI" METEORIITTIA"
  :REM 2E
230 INPUT"<2DOWN>UUSI PELI K/E";VA$:IF VA$<>"K"AND
```

```
   VA$<>"E" THEN 230:REM 92
240 IF VA$="E" THEN END:REM AC
250 RESTORE:REM BB
1000 V=53248: X=150:Y=70:VA=70:REM 6B
1010 FORSI=13*64 TO 14*64-2:REM 75
1020 READ SP:POKESI,SP:NEXT:REM CF
1030 POKEV+21,1:REM 34
1040 POKE2040,13: POKE V+1,Y: POKE V+31,0:REM 94
1050 POKEV+39,10:POKEV+37,5:POKEV+38,7:POKEV+28,1:
  REM 8D
1060 PRINT"<CLR/HOME><25DOWN>":REM D4
1100 POKEV+32,0:POKEV+33,0:GOTO10:REM B3
10000 REM SPRITE 1:REM DD
10010 DATA 0,0,0,0,16,0,0,16,0,0,7,0,0,9,0,0:REM 1
  E
10020 DATA 8,80,0,8,0,0,20,0,0,255,0,1,85,64,5,195
  :REM 73
10030 DATA 80,7,0,208,21,195,84,5,85,80,3,255,192,
  0,85,0:REM BB
10040 DATA 2,20,128,8,20,32,32,0,8,32,0,8,168,0,42
  ,0:REM DO
```

Mato

Tommi Timonen

Mato-pelissä ohjaat nälkäistä matoa, jonka herkkuruokaa ovat numerot. Se saa syötyä niitä eri puolilta näyttöä, johon niitä ilmestyy satunnaisesti aina edellisen tultua syödyksi. Valitettavasti vain mato lihoo sitä enemmän, mitä isomman numeron hän syö. Mitä pidemmäksi mato venyy, sitä suuremmaksi käy riski, että mato puree vahingossa itseään, eikä se suinkaan ole mitään myrkyttömintä lajia. Myös reviirin (näytön) ulkopuolelle joutuminen tietää kuolemaa.

Ohjaus on toteutettu sekä näppäimistöltä että joystickillä.

```
PROGRAM:MATO

5 REM TT 06/04/85:REM 56
10 POKE53280,0:POKE53281,7:PRINT"<CLR/HOME><BLK>":
  REM EB
20 GOTO2000:REM 5D
97 ::REM AA
98 REM ALKURUTIINIT:REM FA
99 ::REM AC
100 DIMP%(1000):REM F2
110 PRINT"<CLR/HOME><BLK><RVS ON><14SPACE>PISTEET:
    000<14SPACE>":REM 80
120 FORA=0TO8:P%(A)=1514+A:POKE1514+A,87:NEXT:REM
  6C
130 P%(9)=1523:POKE1523,81:REM 64
140 E=9:V=0:X=19:Y=11:S=4:REM B6
150 GOSUB990:REM B8
197 ::REM DB
198 REM OHJAUS:REM 50
199 ::REM DD
200 A=PEEK(197):IFA=64THEN250:REM 9E
210 IFA=33THENS=1:REM F6
215 IFA=36THENS=2:REM FF
220 IFA=34THENS=3:REM FA
```

```
225 IFA=37THENS=4:REM 03
240 GOTO290:REM 6A
250 A=(NOTPEEK(56320))AND15:REM 06
260 IFA=0THEN290:REM 9F
270 IFA=1THENS=1:REM C7
275 IFA=2THENS=2:REM CE
280 IFA=4THENS=3:REM CD
285 IFA=8THENS=4:REM D7
290 X=X+(S=3)-(S=4):REM 09
295 Y=Y+(S=1)-(S=2):REM 0C
300 IFX<0ORX>39ORY<0ORY>23THEN1000:REM 7C
310 POKEP%(E),87:E=E+1:IFE=1000THENE=0:REM B3
320 P%(E)=1064+X+40*Y:IFPEEK(P%(E))=87THEN1000:REM
   23
325 IFPEEK(P%(E))>128THENGOSUB900:REM CB
326 POKEP%(E),81:REM 6A
330 IFL=0THENPOKEP%(V),32:V=V+1:IFV=1000THENV=0:RE
  M 77
340 IFL>0THENL=L-1:REM 42
400 GOTO200:REM 5F
897 ::REM E2
898 REM NUMERO:REM 63
899 ::REM E4
900 L=PEEK(P%(E))-176:REM 6E
910 POKEP%(E),81:REM 69
920 SC=SC+L:S$=MID$(STR$(SC),2):REM 1A
930 IFLEN(S$)<3THENS$="0"+S$:GOTO930:REM 49
940 PRINT"<HOME><23RIGHT><RVS ON>"S$:REM A5
950 FORA=54272TO54295:POKEA,0:NEXT:REM 32
960 POKE54296,15:POKE54277,133:POKE54278,133:POKE5
  4276,17:REM DA
970 FORA=LTO0STEP-1:POKE54273,150+5*A:FORR=0TO100:
  NEXT:NEXT:REM BF
980 POKE54272,0:POKE54273,0:POKE54276,0:REM C7
990 Z=INT(RND(1)*960)+1064:IFPEEK(Z)<>32THEN990:RE
  M E3
991 POKEZ,177+INT(RND(1)*8):REM 26
995 RETURN:REM 87
997 ::REM E3
```

```
998 REM LOPPU:REM 1E
999 ::REM E5
1000 FORA=54272TO54295:POKEA,0:NEXT:REM 55
1010 POKE54296,15:POKE54277,131:POKE54278,123:POKE
  54276,33:REM F8
1020 FORR=0TO50:POKE54273,25+INT(RND(1)*5):NEXT:RE
  M 17
1030 POKE54272,0:POKE54273,0:POKE54276,0:REM EA
1040 POKE198,0:REM F2
1050 POKEZ,32:REM E0
1060 IFV<ETHENFORA=VTOE:POKEP%(A),32:NEXT:GOTO2000
  :REM AE
1070 FORA=VTO999:POKEP%(A),32:NEXT:REM 4B
1080 FORA=0TOE:POKEP%(A),32:NEXT:REM C0
2000 PRINT"<HOME><3DOWN><15RIGHT>MATOPELI":REM E7
2010 PRINT"<3DOWN><32RIGHT>I":REM B0
2020 PRINT"<6RIGHT>OHJAA MATOA NAPPAIMILLA  J K":R
  EM D5
2030 PRINT"<32RIGHT>M":REM 83
2040 PRINT"<2DOWN><6RIGHT>TAI JOYSTICKILLA!":REM F
  8
2060 PRINT"<4DOWN><5RIGHT>PAINA VALILYONTIA ALOITT
  AAKSESI!":REM 12
2100 POKE198,0:REM F0
2110 GETA$:IFA$<>""THENRUN100:REM AA
2120 IF(PEEK(56320)AND16)=0THENRUN100:REM 77
2130 GOTO2110:REM C3
```

Starhunter

Jyrki J. J. Kasvi

Ohjaat pientä ilmapalloa ja tehtäväsi on kerätä taivaalta mahdollisimman monta tähteä mahdollisimman pian. Tehtävä on vaikea, sillä taivaalla vaanii erivärisiä pullistelevia möhköjä, joihin törmääminen tietää hengen menoa. Tähdistä saatavat pisteet riippuvat siitä, kuinka kauan edellisen tähden poimimisesta on kulunut.

Ohjaus on toteutettu kakkosporttiin kytketyllä joystickillä. Muista, että liike on toteutettu vektoreilla, eli nopeus kasvaa niin kauan kuin stickiä kyseiseen suuntaan väännetään. Toisaalta suunnan muuttaminen vaatii päinvastaista vääntöä sitä enemmän mitä suurempi alkuperäinen nopeus on.

```
0 GOTO200:REM FB
1 FORSI=0TO999:POKESI+55296,RND(0)*15+1:NEXT:GOTO2
  00:REM A5
10 IF(PEEK(JS)AND1)=0THENSY=SY-.5:REM 19
38 IF(PEEK(JS)AND2)=0THENSY=SY+.5:REM 22
39 IF(PEEK(JS)AND4)=0THENSX=SX-.5:REM 25
40 IF(PEEK(JS)AND8)=0THENSX=SX+.5:REM 1F
50 X=X+SX:Y=Y+SY:REM 28
51 IFX<24THENX=24:SX=-SX:REM 13
52 IFX>231+89THENX=231+89:SX=-SX:REM AE
53 IFY<58THENY=58:SY=-SY:REM 27
54 IFY>229THENY=229:SY=-SY:REM 8A
59 POKEV+1,Y:IFX<256THENPOKEV+16,PEEK(V+16)AND254:
  POKEV,X:GOTO70:REM 29
60 POKEV+16,PEEK(V+16)OR1:POKEV,X-256:REM E6
70 POKEV+29,INT(RND(0)*128)*2:REM 70
80 POKEV+23,INT(RND(0)*128)*2:REM 6B
90 IF(PEEK(V+30)AND1)<>0THEN500: GAME OVER:REM A5
100 IF(PEEK(V+31)AND1)=0THEN10: NO STAR:REM DE
105 PM=500-TI:TI$="000000":IFPM<0THENPM=0:REM 76
110 PI=PI+PM:SC=SC+1:PRINT"<HOME><WHT>SCORE:";PI;:
  REM BO
```

```
120 IFSC>7THEN300:REM EF
130 POKEV+21,PEEK(V+21)OR2↑SC:REM F4
140 MX=INT(RND(0)*320):MY=INT(RND(0)*170+58):IFABS
  (MY-Y)<50ORABS(MX-X)<50THEN140:REM 03
160 POKEV+1+2*SC,MY:IFMX<256THENPOKEV+2*SC,MX:GOTO
  180:REM 5C
170 :POKEV+2*SC,MX-256:POKEV+16,PEEK(V+16)OR2↑SC:R
  EM B1
180 POKEV+39+SC,RND(0)*15+1:GOTO300:REM F9
200 V=53248:X=50:Y=150:JS=56321:REM DA
210 PRINT"<CLR/HOME>";:POKEV+21,1:PRINT"<CLR/HOME>
  <WHT>SCORE:";SC:REM 06
220 POKE2040,13:FORSI=2041TO2047:POKESI,0:POKESI+5
  1247,RND(0)*15+1:NEXT:REM D4
230 POKEV+39,2:POKEV+37,10:POKEV+38,8:POKEV+32,0:P
  OKEV+33,0:REM 00
240 POKEV+28,1:TI$="000000":REM A9
250 FORSI=13*64TO14*64-1:READSD:POKESI,SD:NEXT:REM
   77
300 POKETA,32:TA=RND(0)*920+1070:POKEV+31,0:REM BB
305 POKEV+29,254:POKEV+23,254:REM 90
310 FORSI=1TO10:POKETA,42:IFPEEK(V+31)>0THEN300:RE
  M 6D
320 NEXT:POKEV+29,0:POKEV+23,0:POKEV+30,0:POKEV+31
  ,0:REM 83
330 FORSI=1TO10:IFPEEK(V+31)>0THEN300:REM DF
340 NEXT:GOTO10:REM AA
500 PRINT"<HOME><6DOWN><WHT><15RIGHT>GAME OVER!":I
  NPUT"<3DOWN>UUSI PELI <RVS ON>K<RVS OFF>/<RVS ON
  >E<RVS OFF>";VA$:REM BC
510 IFVA$="K"THENRUN:REM D0
10000 REM SPRITE 1 ← ILMAPALLO:REM D7
10010 DATA 0,166,0,2,166,128,10,165,160,42,165,168
  ,42,169,104,170:REM 6F
10020 DATA 169,106,170,169,90,170,169,90,42,169,88
  ,42,169,104,42,169:REM 31
10030 DATA 104,10,169,96,14,165,176,12,166,48,3,0,
  192,3,0,192:REM B0
10040 DATA 3,0,192,0,255,0,0,255,0,0,255,0,0,60,0,
  0:REM 8C
```

Näppäilytesti

Jyrki J. J. Kasvi

Tämä pieni peli testaa pelaajan näppäimistön tuntemusta. Näytön keskelle tulostuu jostakin merkistä muodostettu rivi, jota vastaavaa näppäintä on painettava mahdollisimman nopeasti. Oikean näppäimen painamisen jälkeen etsittävä merkki vaihtuu. Tämä jatkuu kymmenen sekuntia, jonka jälkeen ohjelma ilmoittaa kuinka monta kirjainta pelaaja tänä aikana ehti löytää.

```
PROGRAM:NAPPAILYNOPEUS

0 PRINT"<CLR/HOME>": LA=RND(-TI):REM 97
10 TI$="000000": LA=0:REM F7
20 OI$=CHR$(INT(RND(0)*27+64)):NI$=OI$+OI$+OI$+OI$
  +OI$+OI$:REM E7
30 PRINT"<HOME><12DOWN><17RIGHT>";NI$:REM F6
40 GETVA$: IFTI>1000 THEN 100:REM 24
50 IF VA$<>OI$ THEN 40:REM 78
60 LA=LA+1: GOTO 20:REM EE
100 PRINT"<2DOWN>SAIT";LA;" KIRJAINTA OIKEIN":REM
  58
110 INPUT"<DOWN>HALUATKO PELATA UUDESTAAN <RVS ON>
  K<RVS OFF>/<RVS ON>E<RVS OFF>";VA$:REM 2E
120 IF VA$="K" THEN 0:REM 08
```

9

Hyötyohjelmat

Funktionäppäin

Kristian Rautiainen

Funktionäppäin-ohjelman avulla voi ohjelmoida kahdeksan omavalintaista toimintoa (komentoa tai käskyä) Commodore 64:n neljälle funktionäppäimelle.

Kun ohjelma ajetaan, näytölle ilmestyy kysymys 'toiminto nappaimelle F-1'. Tällöin tulee kirjoittaa F1-näppäimelle haluttava toiminto. Siinä ei saa kuitenkaan olla enempää kuin 9 merkkiä välilyönnit ym. mukaanluettuna. Se ei myöskään saa sisältää pilkkua (,) tai kaksoispistettä (:). Jos haluaa ohjelmoida esimerkiksi F1:lle RUN-komennon niin, että mikä tahansa Basic-ohjelma käynnistyy vain F1:stä painamalla, tulee kirjoittaa RUN(RETURN). Tällöin tietokone ymmärtää F1-näppäimen painamisen samalla tavalla kuin jos kirjoittaisi RUN ja sitten painaisi RETURN-näppäintä, eli ohjelma käynnistyy (tässä toiminnossa on neljä merkkiä). Samalla tavoin menetellään muiden funktio-näppäinten kohdalla.

Tämän jälkeen tietokone siirtää toiminnot muistiin. Mikäli näytölle ilmestyy virheilmoitus, on annettu toiminto sisältänyt enemmän kuin yhdeksän merkkiä. Sitten ohjelma lataa konekieliohjelman, jonka avulla

juuri annettuja toimintoja voi käyttää.

Mikäli tietokone suljetaan tai resetoidaan tai painetaan RUN/STOP- ja RESTORE-näppäimiä yhtä aikaa tai käytetään komentoa SYS 64738, konekieliohjelma kytkeytyy pois päältä. Tällöin se voidaan käynnistää uudelleen (mikäli virtaa ei ole katkaistu) komennolla SYS 50000.

Basic-ohjelma koostuu seuraavista osista:

7	tyhjentää näytön
10	vastaanottaa näppäintoiminnot
12	lataa pari numeroarvoa muistiin
15 - 40	tarkastavat toiminnot ja siirtävät ne muistiin
60	lataa konekieliohjelman muistiin
62	tarkistaa konekieliohjelman
190	käynnistää konekieliohjelman

Konekieliohjelma on hyvin yksinkertainen. Komennolla SYS 50000 vaihdetaan muistipaikkojen 655 ja 656 arvot niin, että konekieliohjelma käynnistyy joka kerran, kun jotakin nappia painetaan. Tällöin se tarkistaa, onko SHIFT-näppäintä painettu ja sen jälkeen, onko funktionäppäimiä painettu. Jos näin on, kone suorittaa asianomaiselle funktionäppäimelle määritellyn toiminnon. Muuten kone ei tee mitään tavallisuudesta poikkeavaa.

```
PROGRAM:FUNKTIONAPPAIN

0 REM*****************************:REM 58
1 REM*                            *:REM 69
2 REM* BY KRISTIAN RAUTIAINEN     *:REM 5A
3 REM*                            *:REM 6B
4 REM*      HELSINKI, 1985        *:REM C6
5 REM*                            *:REM 6D
6 REM*****************************:REM 5E
7 PRINT"<CLR/HOME>":REM 9B
10 FORA=1TO8:PRINT"TOIMINTO NAPPAIMELLE F";A;:INPU
  TF$(A):NEXT:REM B4
12 C(0)=49132:FORA=1TO8:C(A)=C(A-1)+20:NEXT:REM 97
15 FORA=1TO8:REM D7
20 IFF$(A)=""THEN40:REM 01
25 IFRIGHT$(F$(A),7)="+RETURN"THENF$(A)=LEFT$(F$(A
  ),LEN(F$(A))-7)+CHR$(13):REM 75
```

```
27 IFLEN(F$(A))>9THENPRINT"F";A;"ON LIIAN PITKA, K
  IRJOITA JOTAKIN MUUTA";:GOTO29:REM 9D
28 GOTO30:REM 06
29 INPUTF$(A):GOTO20:REM CC
30 POKEC(A),LEN(F$(A)):REM BF
35 FORZ=1TOLEN(F$(A)):POKEC(A)+Z,ASC(MID$(F$(A),Z,
  1)):NEXT:REM A4
40 NEXT:REM A3
45 REM LADATAAN NAPPAINTOIMINNOT MUISTIIN:REM BD
50 A=50000:REM D8
60 FORQ=0TO232:READD:POKEA+Q,D:X=X+D:NEXT:REM 84
62 IFX<>33771THENPRINT"<2DOWN>DATA ERROR":END:REM
  A5
65 REM LADATAAN KONEKIELIRUTIINI:REM 44
70 DATA169,91,141,143,2,169,195,141,144,2,96:REM B
  2
75 DATA162,0,173,141,2,201,1,208,3,76,204,195:REM
  C9
80 DATA165,197,201,4,240,15,201,5,240,31:REM DO
85 DATA201,6,240,47,201,3,240,63,76,72,235:REM 3E
90 DATA172,0,192,232,189,0,192,157,118,2:REM E3
95 DATA236,0,192,208,244,132,198,76,46,196:REM 5E
100 DATA172,40,192,232,189,40,192,157,118,2:REM 73
105 DATA236,40,192,208,244,132,198,76,46,196:REM B
  A
110 DATA172,80,192,232,189,80,192,157,118,2:REM 7C
115 DATA236,80,192,208,244,132,198,76,46,196:REM B
  F
120 DATA172,120,192,232,189,120,192,157,118,2:REM
  D3
125 DATA236,120,192,208,244,132,198,76,46,196:REM
  EB
130 DATA165,197,201,4,240,15,201,5,240,31:REM FC
135 DATA201,6,240,47,201,3,240,63,76,72,235:REM 6A
140 DATA172,20,192,232,189,20,192,157,118,2:REM 73
145 DATA236,20,192,208,244,132,198,76,46,196:REM B
  C
150 DATA172,60,192,232,189,60,192,157,118,2:REM 7C
155 DATA236,60,192,208,244,132,198,76,46,196:REM C
  1
```

```
160 DATA172,100,192,232,189,100,192,157,118,2:REM
  D3
165 DATA236,100,192,208,244,132,198,76,46,196:REM
  ED
170 DATA172,140,192,232,189,140,192,157,118,2:REM
  DC
175 DATA236,140,192,208,244,132,198:REM F7
180 DATA162,200,160,255,136,208,253,202,208,248,96
  :REM D2
190 SYS50000:REM 8E
200 REM KAYNNISTETAAN KONEKIELIOHJELMA:REM EB
```

Texti Master

Timo Määränen

Texti Master koostuu kahdesta ohjelmasta. Ensiksi on 'SNKAND BC->MC', joka luo levylle konekielisen ohjelman nimeltä 'SET2 & SKANDIT'. Tämä ohjelma muuttaa Kuusnelosen merkistön skandinaaviseksi seuraavaan tapaan:

Å - <shift> + <->
å - <shift> + <]>
ä - <shift> + <[>
Ä - <shift> + <+>
Ö - <comm.> + <-> ö - <punta>

Mikäli merkistönmuunto-ohjelmaa halutaan käyttää sinällään, käy se seuraavasti:

```
LOAD"SET2 & SKANDIT",8,1
SYS 50200
```

Aluksi varsinainen Texti Master -ohjelma lataa merkistönmuunto-ohjelman levyltä ja ajaa sen. Tämän jälkeen skandinaaviset merkit ovat käytössä. Sitten ohjelma kysyy, montako merkkiä tulee riville. Tätä tietoa käytetään myös rivintasausrutiinissa.

IST/DEL ja kursorinäppäimet sivusuntaan toimivat normaalisti. Sen sijaan kursorinäppäin pystysuuntaan siirtää tekstin kuvaruudun ylempään puoliskoon siten, että kuusi edellistä riviä plus nykyinen rivi jäävät näkyviin.

Funktionäppäimillä saadaan seuraavat toiminnot:

F1 — selaus rivinumeroineen.
F3 — selaus rivinumeroitta.
F5 — autotab on/off. Jättää yhdeksän välilyöntiä rivin alkuun.
F7 — tekstin muokkaaminen. Valittavana ovat toiminnot:
0 — muistin nollaus.
1 — poistaa rivi(t).
2 — lisää rivi(t).
3 — siirrä rivi(t).
4 — tasaa rivi(t).
5 — korjaa rivi(t).

6 — paluu normaalin kirjoitustilaan.
F2 — tulostus printterille.
F4 — talletus levylle. Käytettävissä kaksi tapaa:
1 — lisätään vanhan tiedoston perään.
2 — korvataan vanhat tiedot uusilla.
F6 — haku levyltä. Käytettävissä kaksi tapaa:
1 — liitetään uusi teksti vanhan perään.
2 — otetaan uusi teksti vanhan paikalle.
F8 — levyn hakemisto. Kaksi tapaa:
1 — tulostetaan hakemisto näytölle.
2 — tulostetaan hakemisto näytölle JA lisätään se muistissa olevan tekstin perään.

```
PROGRAM:SKAND BC->MC

10 REM ************************************:REM 85
12 REM *                                    *:REM 9B
14 REM *    CHARACTER SET2 & SKANDIT        *:REM 7C
16 REM *                                    *:REM 9F
18 REM *       FOR COMMODORE-64             *:REM C4
20 REM *                                    *:REM 9A
22 REM *      SCREEN 49152-50176            *:REM 91
24 REM *                                    *:REM 9E
26 REM *      SET2    51200-53248           *:REM E9
28 REM *                                    *:REM A2
30 REM *    BY TIMO MAARANEN 12,1984        *:REM 17
32 REM *                                    *:REM 9D
34 REM ************************************:REM 8B
36 ::REM A3
38 ::REM A5
40 SA=50200 : EA=50474 : SU=0:REM 7C
42 HI=INT(SA/256) : LO=SA-HI*256:REM FB
44 FOR T=SA TO EA:REM 9D
46 READ A : POKE T,A :SU=SU+A:REM 24
48 NEXT T:REM FF
50 IF SU<>44946 THEN PRINT" VIRHE":STOP:REM 1F
52 PRINT:PRINT" SAVING...SET2 & SKANDIT" : RESTORE
```

```
  :REM 01
54 OPEN1,8,1,"@0:SET2 & SKANDIT":REM F9
56 PRINT#1,CHR$(LO);CHR$(HI);:REM BE
58 FOR T=SA TO EA:REM A2
60 READ A:REM C3
62 PRINT#1,CHR$(A);:REM 43
64 NEXT T : CLOSE1:REM DE
66 PRINT:PRINT" START: SYS50200":REM C2
68 END:REM 45
80 DATA 173,2,221,9,3,141,2,221,173,0,221,41,252,1
   41,0,221,173,24,208,41,15,141:REM FE
82 DATA 24,208,169,192,141,136,2,173,14,220,41,254
   ,141,14,220,169,251,37,1,133,1:REM 50
84 DATA 169,0,133,251,133,253,169,216,133,252,169,
   200,133,254,162,0,160,0,177:REM CA
86 DATA 251,145,253,200,192,255,208,247,177,251,14
   5,253,232,230,252,230,254,224:REM 39
88 DATA 9,208,232,165,1,9,4,133,1,173,14,220,9,1,1
   41,14,220,173,24,208:REM 62
89 DATA 41,240,9,2,141,24,208,169,147,32,210,255:R
   EM 3B
90 DATA 169,0,141,217,200:REM C5
91 DATA 141,223,200,141,224,200,141,226,200,141,23
   1,200,141,233,200,141,239,200:REM F5
92 DATA 141,223,202,141,231,202,141,233,202,141,23
   9,202,169,54,141:REM 97
94 DATA 216,200,169,60,141,218,200,141,227,200,141
   ,230,200,141,234,200,141,218:REM D9
96 DATA 202,141,225,202,141,230,202,141,234,202,16
   9,6,141,219,200,141,235,200:REM AA
98 DATA 169,62,141,220,200,141,222,200,141,236,200
   ,141,238,200,169,102,141,221:REM DF
99 DATA 200,141,225:REM A5
100 DATA 200,141,228,200,141,229,200,141,237,200,1
   41,216,202,141,219,202,141:REM 69
102 DATA 221,202,141,222,202,141,226,202,141,227,2
   02,141,228,202,141,229,202:REM 6C
104 DATA 141,235,202,141,237,202,141,238,202,169,2
   4,141,232,200,141,232,202,141:REM 08
106 DATA 217,202,169,126,141,220,202,141,236,202,1
```

```
  69,195,141,224,202,96:REM 9F
110 ::REM CC
112 ::REM CE
114 ************************************:REM 54
116 *                                  *:REM EC
118 * OHJELMA LATAA VALMIIN KONEKIE-   *:REM 94
120 * LISEN "SET2 & SKANDIT"-NIMISEN   *:REM 38
122 * OHJELMAN LEVYLLE,JOKA LADATAAN   *:REM DB
124 * KONEESEEN SEURAAVASTI:           *:REM 0A
126 *                                  *:REM ED
128 *  LOAD"SET2 & SKANDIT",8,1        *:REM 66
130 *                                  *:REM E8
132 *  SYS 12288                       *:REM EE
134 *                                  *:REM EC
136 *  NEW                             *:REM D8
138 *                                  *:REM F0
140 * VAROITUS:                        *:REM A0
142 *                                  *:REM EB
144 * <RUN/STOP-RESTORE> KIELLETTY     *:REM D9
146 *                                  *:REM EF
152 * KONE EI HYYDY VAAN KUVARUUTU     *:REM 40
154 * MUUTTAA TAKAISIN VANHALLE PAI-   *:REM B5
156 * KALLLE, MUTTA SCREEN EDITOR EI!  *:REM 92
158 * PELASTUS:                        *:REM 9D
160 *                                  *:REM EB
162 *  POKE 648,4  TAI SYS50200        *:REM F2
164 ************************************:REM 59


PROGRAM:TEXTIMASTER

0 REM ************************************:REM 7E
1 REM *                                  *:REM 69
2 REM * TEXTI MASTER  (C) T.MAARANEN     *:REM 1D
3 REM *                                  *:REM 6B
4 REM ************************************:REM 82
5 PRINTCHR$(147);:REM EB
6 IF YU=0 THEN YU=1:LOAD"SET2 & SKANDIT",8,1:REM D
  C
7 SYS 50200:DIMQ$(405):DIMFF$(30):REM 18
```

```
8 FOR T=0 TO 7:READ A:POKE51976+T,A:NEXTT:REM 7F
9 FOR T=0 TO 7:READ A:POKE51984+T,A:NEXTT:GOTO14:R
  EM 57
10 DATA 0,0,24,0,0,24,0,0,0,0,0,0,0,12,12,24:REM 1
  7
14 PRINT" ________ ______ __<4SPACE>__ ________ -
  ":REM 3E
15 PRINT" |   |   |K4SPACE>   K4SPACE>|   |   |
   |  | ":REM 3E
16 PRINT"     |    K7SPACE> K4SPACE>|     |     |
  | ":REM BB
17 PRINT"<5SPACE> K4SPACE> K7SPACE> |___| <5SPACE> |
  <5SPACE> | ":REM 93
18 PRINT"<5SPACE> K4SPACE> |--<6SPACE> | <7SPACE> K
  5SPACE> | ":REM EB
19 PRINT"<5SPACE> K4SPACE> K7SPACE> |---| <5SPACE> |
  <5SPACE> | ":REM A6
20 PRINT"<5SPACE> K4SPACE> K7SPACE> K4SPACE>| <5SPA
  CE> K5SPACE> | ":REM C2
21 PRINT"<5SPACE> K4SPACE> K4SPACE>   K4SPACE>| <5
  SPACE> K5SPACE> | ":REM 72
22 PRINT"<4SPACE> --   ----- --<4SPACE>--    --
     - ":REM 88
23 PRINT:REM F2
24 PRINT" __ __ ___ ___ ______ ____ ____":RE
  M 7D
25 PRINT" | || | K3SHIFT+SPACE>| |  | |  |  | K5S
  PACE> K4SPACE>| ":REM 20
26 PRINT" | || | K3SHIFT+SPACE>| K7SPACE> |    K5
  SPACE> K4SPACE>| ":REM 0F
27 PRINT" |  -  | K3SHIFT+SPACE>| |__<4SPACE> |
   |__   |___| ":REM 4B
28 PRINT" K5SPACE>| |--| <4SPACE>|    |    K5SPACE
  > |   | ":REM A9
29 PRINT" K5SPACE>| K3SHIFT+SPACE>| <4SPACE>|    |
      K5SPACE> |   | ":REM 65
30 PRINT" K5SPACE>| |  | |  |   |    K5SPACE> |
   | ":REM 27
31 PRINT" K5SPACE> K3SHIFT+SPACE>  ---<4SPACE> -
  <4SPACE>---   -   - ":REM 56
```

```
32 PRINT:PRINT"<GREY3><5SPACE>(-) |╮╲╭<SHIFT+SPACE
  >╲♠♠_♠╱‾╱  4,1985 <LT.BLUE>":REM 99
33 PRINT:PRINT"<4SPACE>╯+||‾||+×╮♥♥+<SHIFT+SPACE>╭
  ╱<SHIFT+SPACE>400 _╮×╮+.":PRINT:REM 34
34 PRINT"<4SPACE>╲╭╱|♠<SHIFT+SPACE>╲‾_╯╯╮+╱_╮×╮ (3
  0-80) ?";:MIN=30:MAX=80:GOSUB 50:MR=NU:REM FC
35 TA$="":CV$="":REM 1A
36 PRINTCHR$(147);"<6SPACE>♠L♠ |♠╱ ╯╮_╰╭╮||♠♠<SHIFT
  +SPACE>!!":PRINT:GOTO300:REM A9
40 IF R<6 OR FLAG=1 THEN RETURN:REM EF
42 PRINTCHR$(147);:REM 1C
43 FOR KE=R-5 TO R:N$=Q$(KE):P=LEN(N$):REM 6C
44 IF LEFT$(N$,1)="|" THEN PRINT"<9SPACE>";RIGHT$(
  N$,P-1) : GOTO 46:REM F3
45 PRINT N$:REM 68
46 NEXT KE:REM 39
47 PRINT TA$;E$; : RETURN:REM AA
50 POKE204,0 :OSA$="":REM 48
51 GET LUKU$ :IF LUKU$="" THEN 51:REM EF
52 E=ASC(LUKU$):IF E=20 THEN P=LEN(OSA$):GOTO60:RE
  M EF
53 IF E=13 THEN58:REM 79
54 IF E<48 OR E>57 THEN 51:REM 0A
55 IFLEN(OSA$)>2 THEN 51:REM 35
56 OSA$=OSA$+LUKU$:PRINTLUKU$;:GOTO51:REM 86
58 NU=VAL(OSA$):IF NU<MIN OR NU>MAX THEN 51:REM 11
59 POKE204,1:PRINT" <LEFT> <LEFT>":RETURN:REM EF
60 IF P>1THENOSA$=LEFT$(OSA$,P-1):POKE204,1:PRINT"
   <2LEFT> <LEFT>";:POKE204,0:GOTO51:REM 55
61 IFLEN(OSA$)=1 THENOSA$="":POKE204,1:PRINT" <2LE
  FT> <LEFT>";:GOTO50:REM A3
62 GOTO 51:REM 07
65 GETB$:IF B$="" THEN RETURN:REM 70
66 GET B$:IF B$=CHR$(13) THEN T=R-1:RETURN:REM 5E
67 IF B$<>"" THEN RETURN:REM 2F
68 GOTO 66:REM 13
69 :  REM * * * INPUT RUTIINI * * * * * *:REM BB
70 RV=MR:E$="":EA$="":X=1:PRINT TA$;:POKE204,0:REM
   8A
72 GETEA$:IF EA$="" THEN 72:REM 7F
```

```
74 E=ASC(EA$):IF E<>13 THEN 80:REM 42
75 IF E=13 THEN POKE204,1:IF X<LEN(E$)+1 THENPRINT
  MID$(E$,X,1):RETURN:REM F3
76 IF E$="" THENPRINT"<LEFT>  ":RETURN:REM 9D
77 PRINT" <LEFT> <LEFT>":RETURN:REM 93
80 IF E=20 OR E=148 OR E=29 OR E=157 THEN 101:REM
  AA
81 IF E=58 THEN E=161:GOTO89:REM 14
82 IF E=17 OR E=145 THEN POKE204,1 :GOSUB 40 :POKE
  204,0:REM 6A
83 IF E=160 THEN E=32:REM 29
84 IF E=44 THEN E=162:GOTO89:REM 13
85 IF E>127 AND E<193 THEN 135:REM D2
88 IF E<31 OR E>221 THEN EA$="":GOTO72:REM D3
89 P=LEN(E$):IF P>=RV AND X>P THEN 72:REM 24
90 IF X>P THEN E$=E$+CHR$(E):PRINTEA$;:X=X+1:GOTO
   72:REM E9
91 IF X<=P THEN 93:REM B5
92 GOTO72:REM 0D
93 V$=LEFT$(E$,X-1):IF P-X<1 THEN O$="":REM 3C
94 IF P-X>0 THEN O$=RIGHT$(E$,P-X):REM 7B
98 E$=V$+CHR$(E)+O$:PRINTEA$;:X=X+1:GOTO72:REM FC
101 IF E=29 OR E=157 THEN 250:REM 94
102 IF E=148 THEN 116:REM 08
103 P=LEN(E$):REM BA
104 IF P=0 THEN 72:REM 79
106 IF P=1 AND X=2THEN E$="":PRINT" <2LEFT> <LEFT>
  ";:X=X-1:GOTO 72:REM 15
107 IF X>P THEN E$=LEFT$(E$,P-1):PRINT" <2LEFT> <L
  EFT>";:X=X-1:GOTO72:REM 43
108 IF X<=P AND X>=2 THEN POKE204,1:GOTO112:REM 79
109 IF X=2 AND P=2 THEN E$=RIGHT$(E$,1):PRINT"<LEF
  T>";E$;" <2LEFT>";:X=1:GOTO72:REM 2C
110 GOTO 72:REM 34
112 V$=LEFT$(E$,X-2):O$=RIGHT$(E$,P-X+1):E$=V$+O$:
  PRINT"<LEFT>";O$;" ";:X=X-1:REM 67
113 FOR T=1 TO LEN(O$)+1:PRINT"<LEFT>";:NEXTT:POKE
   204,0:GOTO 72:REM C7
116 P=LEN(E$):REM BE
117 IF X>0 AND X<P+1 AND P<RV THEN 120:REM CA
```

```
118 GOTO 72:REM 3C
120 POKE204,1:V$=LEFT$(E$,X-1):O$=RIGHT$(E$,P+1-X)
  :E$=V$+" "+O$:REM 9B
123 PRINT" ";O$;:FORT=1TOLEN(O$)+1:PRINT"<LEFT>";:
  NEXTT:REM 85
124 POKE 204,0:GOTO72:REM 94
126 IF E=29 THEN 131:REM D9
128 IF X=1 THEN 72:REM 88
129 X=X-1:PRINT"<LEFT>";:GOSUB150:GOTO72:REM F6
131 IF X=X+1 THEN 72:REM 05
132 X=X+1:PRINT"<RIGHT>";:GOSUB150:GOTO72:REM 6E
133 GOTO 72:REM 39
135 IF E>132 AND E<141 AND FLAG=0 THEN 140:REM 49
136 IF FLAG=1 AND (E<17 OR E>20) AND (E<145 OR E>1
  48) THEN 89:REM E4
138 GOTO 72:REM 3E
140 IF E=137 THEN160:REM 07
141 IF E=138 THEN 325:REM 0C
142 IF E=139 THEN 359:REM 15
143 IF E=133 THEN 400:REM 03
144 IF E=134 THEN 412:REM 08
145 IF E=135 THEN 422:REM 0B
146 IF E=136 THEN POKE204,1:GOTO 430:REM A1
147 IF E=140 THEN 600:REM 07
150 GOTO 72:REM 38
158 REM* ESIMERKKI PRINTTERIN KÄYTöSTÄ *:REM 3A
160 PRINTCHR$(147)"<RVS ON>| ⅃⅂Γ●| ⁄●<RVS OFF> ⅂L⌝⁄|
  |‾_⌝LL‾":PRINT:PRINT" _IVEJÄ";R:PRINT:REM AC
161 PRINT" ♠LOITUSRIVI  ><4SPACE>LOPETUSRIVI   >":
  PRINT:REM 6D
162 PRINT" ●IVUN PITUUS >":PRINT:REM 6E
164 PRINT"<HOME><3DOWN>":PRINTTAB(15);:MIN=0:MAX=R
  :GOSUB 50:IF NU=0 THEN A$="E":GOTO173:REM DC
165 AL=NU:REM 09
166 PRINTTAB(34)"<UP>";:MIN=AL:MAX=R:GOSUB 50:LO=N
  U :PRINT:REM FB
167 PRINTTAB(15);:MIN=37:MAX=400:GOSUB 50:LE=NU:RE
  M 74
171 PRINT:PRINT" ΓLETKO VARMA (K/E) ?";:REM 80
172 GET A$: IF A$="" THEN 172:REM 57
```

```
173 IF A$="E" THEN PRINTCHR$(147);E$;:POKE204,0:GO
  TO 72:REM 15
174 IF A$="K" THEN 200:REM 1D
175 GOTO 172:REM 70
200 OPEN4,4,7:REM BB
202 FOR A=AL TO LO :B$=Q$(A):IF B$="" THEN B$=" ":
  REM 6E
203 N=(A-AL)/LE:IFN=1ORN=2ORN=3ORN=4ORN=5ORN=6ORN=
  7ORN=8ORN=9ORN=10ORN=11THEN240:REM 9D
204 PRINT#4,"<9SPACE>";:FOR I=1 TO LEN(B$):REM 36
205 X$=MID$(B$,I,1):B=ASC(X$):REM 34
206 IF X$="|" THEN PRINT#4,"<9SPACE>";:GOTO 220:RE
  M 9E
209 IF B=161 THEN B=58: GOTO 219:REM 6B
210 IF B=162 THEN B=44: GOTO 219:REM 5F
219 PRINT#4,CHR$(B);:REM 7B
220 NEXTI:PRINT#4:NEXTA:PRINT:PRINT" |ULOSTETTU RI
  VIT";AL;"..";LO:REM 7A
221 PRINT:PRINT"<13SPACE><RVS ON> <_¯| ⁄_⁄> <RVS OF
  F>": E$="":EA$="":REM B4
222 GET A$:IF A$<>CHR$(13) THEN 222:REM FE
223 PRINTCHR$(147):CLOSE4:GOTO70:REM DO
240 PRINTCHR$(147)"<RVS ON>| ⁄⊔⌐●| ⁄●<SHIFT+SPACE>¬⊔
  ⁄|| ¯_⁄⊔⊔¯<19SPACE><RVS OFF>":REM CA
241 PRINT:PRINT" |ULOSTETTU RIVIT";AL;"...";A-1:RE
  M 15
242 PRINT:PRINT" ●IVUN VAIHTO :":PRINT:PRINT" ×AIH
  DA PAPERI ; <_¯| ⁄_⁄> KUN VALMIS":REM 59
243 GET A$:IFA$<>CHR$(13) THEN 243:REM 04
244 PRINT:PRINT"<14SPACE>KIRJOITAN...":GOTO 204:RE
  M 70
250 P=LEN(E$):IF P=0 THEN 72:REM DB
251 IF E=29 THEN 257:REM E1
252 IF X=1 THEN 72:REM 86
253 POKE 204,1:REM BC
254 IF X=P+1 THEN PRINT" <2LEFT>";:POKE204,0:X=X-1
  :GOTO 72:REM 9C
255 IF X>1 AND X<P+1 THEN PRINTMID$(E$,X,1);"<2LEF
  T>";:POKE 204,0:X=X-1:GOTO 72:REM 0A
256 GOTO 72:REM 3F
```

```
257 IF X=P+1 THEN 72:REM 06
258 POKE204,1:REM C1
259 IF X<=P  THEN PRINTMID$(E$,X,1);:POKE204,0:X=X
  +1:GOTO 72:REM 9A
260 GOTO 72:REM 3A
300 GOSUB 70:P=0:REM 71
301 IF R>=400 THEN 312:REM 49
303 R=R+1:Q$(R)=CV$+E$:REM B3
312 IF R>=400 THEN PRINT:PRINT" ┌LET KÄYTTÄNYT KAI
  KEN TILAN.":GOTO 320:REM 5F
316 GOTO300:REM 66
320 PRINT:PRINT" ¬HDOTAN ETTÄ TALLETAT TEKSTIN LEV
  YLLE ":REM 92
321 PRINT:PRINT" |OINEN VAIHTOEHTO ON JOKO TULOSTU
  S":REM D9
322 PRINT:PRINT" TAI TUHOAMINEN" :GOTO300:REM 0B
324 REM * * * * * * -SAVE- * * * * * * *:REM 28
325 PRINTCHR$(147);"<RVS ON>|♠LL¬| ♣<SHIFT+SPACE>¬
  ╲♥╱¬|╲LL¬<21SPACE><RVS OFF>":REM 30
326 PRINT" ♥INULLA ON KAKSI MAHDOLLISUUTTA :":PRIN
  T:MIN=0:MAX=2:REM 5F
327 PRINT" 1) ♠TT╱¯ - LISÄTÄÄN VANHAN TIEDOSTON":
  PRINT"<13SPACE>PERÄÄN.":REM E0
328 PRINT" 2) ╱¯_╲♠L - KIRJOITETAAN 'UUSI'  TIE-":
  PRINT"<13SPACE>DOSTO.":REM A9
329 PRINT:PRINT"  |╱╲! ╱¯_╲♠ALILLA TAVALLA VOIT AN
  TAA":PRINT"<7SPACE>TIEDOSTOL";:REM 65
330 PRINT"LE VANHAN TIEDOSTON":REM 6D
331 PRINT"<7SPACE>NIMEN. |ÄLLöIN VANHA TIEDOSTO":P
  RINT"<7SPACE>SAA UUDEN SISÄL";:REM D2
332 PRINT"LöN.":PRINT:PRINT"<7SPACE>♠TT╱¯  PUOLES
  TAAN LISÄÄ UUDET":PRINTTAB(7);:REM 99
333 PRINT"TIEDOT VANHOJEN PERÄÄN.":PRINT:PRINT" ╲I
  NKÄ TAVAN VALITSET (1/2) ? ";:REM C4
334 GOSUB50:IF NU=0 THEN PRINTCHR$(147);E$;:POKE20
  4,0:GOTO72:REM 18
335 PRINT:INPUT" |IEDOSTON NIMI";NI$:REM 92
336 IF LEN(NI$)>16 THEN NI$=LEFT$(NI$,16):REM DO
337 IF NU=1 THEN OPEN3,8,3,"0:"+NI$+",S,A":GOTO 34
  2:REM BB
```

```
340 OPEN3,8,3,"@0:"+NI$+",S,W":REM 30
342 IF LEN(E$)>0 THEN R=R+1:Q$(R)=E$:E$="":REM B7
343 PRINT:PRINT"<10SPACE>/IRJOITAN...":PRINT:REM A
  E
344 FOR A=1 TO R:REM 26
346 IF Q$(A)="" THEN Q$(A)=" ":REM 6B
348 PRINT#3,"*";Q$(A):NEXTA:REM 18
350 CLOSE3:REM 41
351 PRINT"<2UP><13SPACE><RVS ON><_¯| ,_/><RVS OFF><
  6SPACE>":REM 8A
352 GETA$:IF A$<>CHR$(13) THEN 352:REM 06
353 PRINTCHR$(147):GOTO70:REM F0
355 ::REM D7
356 REM * * * * * -LOAD- * * * * * * * *:REM 1E
359 PRINTCHR$(147);"<RVS ON> |♠/ ,<RVS OFF> ¯\♥/¯|\L
  |+":PRINT:REM 37
360 PRINT" |AAS ON KAKSI MAHDOLLISUUTTA :":PRINT:P
  RINT" 1) LEVYTIEDOSTO LIITE";:REM 96
361 PRINT"TÄÄN MUISTISSA":PRINT"<4SPACE>OLEVAN TEK
  STIN PERÄÄN NIIN  PIT-":REM AA
362 PRINT"<4SPACE>KÄLTI KUIN MAHTUU.":PRINT:PRINT"
   2) \UISTISSA OLEVA TEKSTI  TUH";:REM 1A
363 PRINT"OTAAN":PRINT"<4SPACE>JA  LEVYTIEDOSTO OT
  ETAAN SEN TI-":PRINT"<4SPACE>LALLE":REM 52
364 PRINT:PRINT"  |/¯\! _IVIN PITUUS MÄÄRÄYTYY  HAU
  N":PRINT"<7SPACE>JÄLKEEN PIS";:REM 02
365 PRINT"IMMÄN KONEEN MUIS-":PRINT"<7SPACE>TISSA
  OLEVAN RIVIN MUKAAN.":REM ED
366 PRINT:PRINT" MINKÄ TAVAN VALITSET ? ";:MIN=0:M
  AX=2:GOSUB 50:IF NU=0 THEN385:REM 02
367 IF NU=1 THEN 369:REM 11
368 R=0:REM 60
369 PRINT:INPUT" |IEDOSTON NIMI";NI$:REM 99
370 IF LEN(NI$)>16 THEN NI$=LEFT$(NI$,16):REM CE
371 OPEN3,8,3,"0:"+NI$+",S,R":REM EF
374 R=R+1:INPUT#3,Q$(R):IF R>399 THEN 382:REM 09
375 Q$(R)=RIGHT$(Q$(R),LEN(Q$(R))-1):REM D1
380 IF LEN(Q$(R))>MR THEN MR=LEN(Q$(R)):REM A2
381 IF ST<>64 THEN 374:REM 83
382 CLOSE3:PRINT"<UP>";:REM 1D
```

```
383 PRINT" ⁄ONEEN MUISTISSA";R;"<LEFT>  RIVIÄ":PRI
  NT" _⌝⊨";MR:PRINTTAB(55)"<RVS ON><_⌝| ⁄_⁄><RVS OF
  F>":REM EC
384 GETA$:IF A$<>CHR$(13) THEN 384:REM 10
385 PRINTCHR$(147);E$;:POKE204,0:GOTO72:REM 31
400 PRINTCHR$(147):IF LEN(E$)>2 THEN R=R+1:Q$(R)=E
  $:REM 45
402 E$="":EA$="":POKE204,1:REM 41
404 FOR T=1 TO R:P=LEN(Q$(T)):P$=LEFT$(Q$(T),1):RE
  M 49
406 PRINT"L⁄⁄‾";T:REM 22
407 IF P$="| " THEN PRINT"<9SPACE>";RIGHT$(Q$(T),P-
  1):PRINT:GOTO410:REM C5
408 PRINTQ$(T):PRINT:REM 0A
410 GOSUB65:NEXTT:GOTO70:REM 27
412 PRINTCHR$(147):IF LEN(E$)>2 THEN R=R+1:Q$(R)=E
  $:REM 48
414 E$="":EA$="":POKE204,1:REM 44
415 FOR T=1 TO R:P=LEN(Q$(T)):P$=LEFT$(Q$(T),1):RE
  M 4B
417 IF P$="| " THEN PRINT"<9SPACE>";RIGHT$(Q$(T),P-
  1):GOTO420:REM 00
418 PRINTQ$(T);" <LEFT>":REM 60
420 GOSUB65:NEXTT:GOTO70:REM 28
422 IF LEN(E$)>2 THEN R=R+1:Q$(R)=E$:REM 94
423 POKE204,1:PRINT" <LEFT>":REM 63
425 IF CV$="| " THEN CV$="":TA$="":MR=MR+9:GOTO70:R
  EM F1
426 IF CV$="" THEN CV$="| ":TA$="<9SPACE>":MR=MR-9:
  GOTO 70:REM F4
427 GOTO 70:REM 3D
428 REM * * * * * KORJAUS * * * * * *:REM 6F
430 PRINTCHR$(147);"<RVS ON>|‾⁄♦|⁄⁄ ⁄⌐⁄⁄♠ ⁄♠<24SHI
  FT+SPACE><RVS OFF>":REM 47
432 PRINT"1=POISTA RIVI(T) ><5SPACE>>":PRINT"2=LIS
  ÄÄ  RIVI(T) ><5SPACE>>":REM 76
433 PRINT"3=SIIRRÄ RIVI(T) ><5SPACE>>":PRINT"4=TAS
  AA  RIVI(T) ><5SPACE>>":REM 5B
434 PRINT"5=KORJAA RIVI(T) ><5SPACE>>":PRINT"6=EXI
  T":REM A9
```

```
435 PRINT"0=NOLLAA MUISTI":REM 6C
438 GET A$:IF A$="" THEN 438:REM 61
439 E=ASC(A$):IF E<48 OR E>54 THEN 438:REM C0
440 IF A$="6" THEN PRINTCHR$(147):GOTO70:REM C7
441 IF A$="0" THEN 450:REM 06
442 IF A$="1" THEN 454:REM 0C
443 IF A$="2" THEN 460:REM 0B
444 IF A$="3" THEN 474:REM 12
445 IF A$="4" THEN 530:REM 0D
446 IF A$="5" THEN FLAG=1:GOTO 506:REM 0D
448 GOTO 438:REM 78
450 FOR K=1 TO R:Q$(K)="":NEXTK:R=0:PRINTCHR$(147)
  "<RVS ON>╱♠╲╱╱╲<SHIFT+SPACE>_╲×╲|<SHIFT+SPACE>| ╱
   ┌|| ╱ <RVS OFF>":REM 13
451 PRINT:GOTO 70:REM 01
454 PRINT"<HOME><DOWN>":PRINTTAB(18);:MIN=0:MAX=R:
  GOSUB 50 :IF NU=0 THEN 430:REM 1D
455 AL=NU:PRINTTAB(24)"<UP>";:MIN=AL:MAX=R:GOSUB50
  :LO=NU:REM DB
457 FOR K=0 TO R-LO:Q$(AL+K)=Q$(LO+K+1):NEXT K:R=R
  -LO+AL-1:GOTO430:REM 36
460 PRINT"<HOME><2DOWN>":PRINTTAB(18);:MIN=0:MAX=R
  -1:GOSUB50:AL=NU:REM 91
463 PRINTTAB(23)"<UP>";AL+1:PRINTTAB(23)"<UP>>":LO
  =AL+1:REM 12
464 PRINTTAB(30)"<2UP><WHT>MONTA?<LT.BLUE>":PRINTT
  AB(29)">";:MIN=0:MAX=401-R:GOSUB50:REM AB
465 IF NU=0 THEN 430:REM 04
466 M=NU :FOR K=R TO AL+1 STEP -1:Q$(K+M)=Q$(K):NE
  XTK :PRINT"<6DOWN>":R=R+M:REM C7
468 FOR K=AL+1 TO AL+M:REM A2
470 PRINT"L╲♥+||+×+ _╲×╲";K:GOSUB70:Q$(K)=CV$+E$:N
  EXTK:GOTO430:REM 6B
474 PRINT"<HOME><3DOWN>":PRINTTAB(18);:MIN=0:MAX=R
  :GOSUB50:IF NU=0 THEN 430:REM 41
476 AL=NU:PRINTTAB(24)"<UP>";:MIN=AL:MAX=R:GOSUB50
  :LO=NU:REM DE
478 PRINTTAB(30)"<2UP><WHT>MINNE?<LT.BLUE>":PRINTT
  AB(29)">";:MIN=0:MAX=R:GOSUB50:M=NU:REM 4D
479 IF M=AL OR M=0 OR M=LO+1 OR(M>AL AND M<LO) OR
```

```
  M=LO THEN 430:REM 54
480 IF LO-AL+1>401-R THEN 497:REM 01
481 IF M<AL AND LO-AL+1<401-R THEN 490:REM E2
482 FOR K=1 TO LO-AL+1:Q$(R+K)=Q$(AL+K-1):NEXT K:R
  EM 82
484 FOR K=1 TO LO-AL+1:Q$(R+K)=Q$(AL+K-1):NEXT K:R
  EM 84
485 FOR K=0 TO M-LO-1:Q$(AL+K)=Q$(LO+K+1):NEXT K:R
  EM 8D
486 FOR K=0 TO LO-AL:Q$(AL+M-LO-1+K)=Q$(R+K+1):NEX
  T K:REM C5
488 GOTO 430:REM 74
490 FORK=1 TO LO-AL+1:Q$(R+K)=Q$(AL+K-1):NEXT K:RE
  M 81
492 FOR K=AL-1 TO M STEP-1:Q$(K+LO-AL+1)=Q$(K):NEX
  T K:REM BC
494 FOR K=0 TO LO-AL:Q$(M+K)=Q$(R+K+1):NEXT K:REM
  E6
496 GOTO 430:REM 73
497 PRINT"<7DOWN> ×DISITKO SUORITTAA SIIRTO-OPERAA
  TIOT"::REM D9
498 PRINT" PIENEMMISSÄ JAKSOISSA, ESIM";401-R;"KER
  -":PRINT" RALLAAN":REM B0
500 PRINT:PRINT"<14SPACE><_¯| ⁄_⁄>":REM 65
502 GET A$:IF A$<>CHR$(13) THEN 502:REM 00
503 GOTO 430:REM 68
506 PRINT"<HOME><5DOWN>":PRINTTAB(18);:MIN=0:MAX=R
  :GOSUB 50:AL=NU:REM 67
508 IF NU=0 THEN FLAG=0 :GOTO 430:REM FC
510 PRINTTAB(24)"<UP>";:MIN=AL:MAX=R:GOSUB 50:LO=N
  U:REM 2C
512 PRINTCHR$(147):FOR K=AL TO LO:REM 87
514 E$=Q$(K):X=LEN(E$)+1:PRINT"└⁄¯";K:PRINTE$;:PO
  KE204,0:GOSUB 72:REM B7
516 Q$(K)=E$:NEXTK:REM 17
518 FLAG=0:GOTO 430:REM 2F
530 PRINT"<HOME><4DOWN>":PRINTTAB(18);:MIN=0:MAX=R
  :GOSUB 50:IF NU=0 THEN 430:REM 4B
532 AL=NU:PRINTTAB(24)"<UP>";:MIN=AL:MAX=R:GOSUB 5
  0:LO=NU:PRINTCHR$(147):REM 8C
```

```
534 PRINT" ⁄+_♥⌝×+LL⌝♥ | || | +!":PRINT:REM 83
536 FOR K=AL TO LO :RV=MR:REM 96
537 IF LEFT$(Q$(K),1)="| " THEN Q$(K)=RIGHT$(Q$(K),
  LEN(Q$(K))-1):SIGN=1:RV=MR-9:REM BB
538 IF LEN(Q$(K))<0.8*RV THEN 544:REM E0
539 IF LEFT$(Q$(K),1)="| " THEN Q$(K)=RIGHT$(Q$(K),
  LEN(Q$(K))-1):SIGN=1:REM 99
540 GOSUB 2000:REM DB
544 IFLEFT$(Q$(K),1)="| "THENPRINT"<9SPACE>";RIGHT$
  (Q$(K),LEN(Q$(K))-1):GOTO548:REM 8E
546 PRINT Q$(K):REM 3D
548 NEXT K:REM 2B
550 PRINT:PRINT"<13SPACE><_¯| ⁄_⁄>":REM 6A
552 GET A$:IF A$<>CHR$(13) THEN 552:REM 0A
554 GOTO 430:REM 6E
600 PRINTCHR$(147):PRINT" ¯⌝_ - DISKETIN HAKEMISTO
  ":REM A8
602 PRINT:PRINT" ♥INULLA ON KAKSI MAHDOLLISUUTTA :
  ":REM 42
604 PRINT:PRINT" 1) ¯IRECTORY TULOSTETAAN KUVARUUD
  ULLE":PRINT:REM 85
606 PRINT" 2) ¯IRECTORY TULOSTETAAN KUVARUUDULLE":
  PRINT"<4SPACE> ⌝♠ LIITETÄÄN MUUN";:REM DF
608 PRINT" MUISTISSA OLEVAN":PRINT"<4SPACE>TEKSTIN
   PERÄÄN.":REM 3C
610 PRINT:PRINT:PRINT" ⁄UMMAN TAVAN VALITSET ? ";:
  MIN=1:MAX=2:GOSUB 50:REM D9
612 IF NU=1 THEN K=0:GOTO 616:REM 30
614 IF NU=2 THEN K=1:REM 24
616 PRINTCHR$(147):IF LEN(E$)>0 THEN R=R+1:Q$(R)=E
  $:E$="":REM 70
617 GOSUB 1000:REM DF
618 PRINT:PRINT"<16SPACE><_¯| ⁄_⁄>":REM 91
620 GET A$: IF A$<>CHR$(13) THEN 620:REM 02
622 PRINTCHR$(147):GOTO 70:REM EF
1000 OPEN15,8,15,"IO":OPEN3,8,3,"#":T=18:S=0:REM 5
  B
1002 PRINT#15,"U1 3 0";T;S:REM 4F
1004 GET#3,T$,S$:REM 42
1006 IF T$="" THEN T$=CHR$(0):REM B5
```

```
1008 IF S$="" THEN S$=CHR$(0):REM B5
1010 T=ASC(T$):S=ASC(S$):REM 5C
1012 PRINT#15,"B-P 3 144":REM D5
1014 FOR N=1 TO 23:GET#3,N$:IF N$="" THEN N$=CHR$(
  0):REM 9B
1015 IF ASC(N$)>127 THEN N$=CHR$(ASC(N$)-128):REM
  5A
1016 E$=E$+N$ :NEXTN:YP=0:REM 8B
1018 IF K=1 THEN GOSUB 1150:REM 88
1020 PRINTE$:E$="":PRINT:IF T=0 THEN 1070:REM 25
1022 PRINT:PRINT"/*\~<13SPACE>| |T~  |_  *~ L~/|| |"
  :REM C0
1024 PRINT"-----------------------------------":
  REM 19
1026 PRINT#15,"U1 3 0";T;S:REM 55
1028 PRINT#15,"U1 3 0";T;S:REM 57
1032 FOR LA=0 TO 7:REM 81
1034 PRINT#15,"B-P 3";LA*32+2:REM F4
1036 GET#3,TY$,TR$,SE$:IF TY$="" THEN 1056:REM F1
1037 GOSUB 1200:REM 0E
1038 PRINT#15,"B-P 3";LA*32+30:REM 29
1040 GET#3,PI$,PH$:GOSUB 1300:REM 4A
1042 PRINT#15,"B-P 3";LA*32+5:REM F6
1044 E$="": FOR N=1 TO 16:REM 9A
1046 GET#3,N$ : IF ASC(N$)>128 THEN N$=CHR$(ASC(N$
  )-128):REM 6D
1047 IF N$="" THEN N$=CHR$(0):REM AE
1048 E$=E$+N$ : NEXT N:REM 40
1050 GOSUB 1100 : REM FILETYPE:REM 88
1052 IF K=1 THEN GOSUB 1150:REM 86
1054 PRINTE$;" ";TY$;" ";TR$;" ";SE$;"   ";PI$:REM
  BC
1056 NEXT LA:REM 98
1060 PRINT#15,"B-P 3 0";0:REM DA
1062 GET#3,T$,S$:REM 46
1064 IF T$="" THEN T$=CHR$(0):REM B9
1066 IF S$="" THEN S$=CHR$(0):REM B9
1068 T=ASC(T$):S=ASC(S$):REM 69
1070 IF T=0 THEN CLOSE3:CLOSE15:RETURN:REM 20
1072 GOTO 1022:REM C8
```

```
1100 YP=ASC(TY$):REM A1
1102 IF YP=129 OR YP=129+64 THEN TY$=" SEQ":RETURN
  :REM 11
1104 IF YP=130 OR YP=130+64 THEN TY$=" PRG":RETURN
  :REM 03
1106 IF YP=131 OR YP=131+64 THEN TY$=" USR":RETURN
  :REM 18
1108 IF YP=132 OR YP=132+64 THEN TY$=" REL":RETURN
  :REM 05
1109 RETURN : REM -VARMUUDEN VUOKSI-:REM BB
1150 IF R>354 THEN PRINT" EI TILAA":RETURN:REM 95
1151 R=R+1:IF YP=0 THEN Q$(R)=E$:R=R+1:Q$(R)=" ":R
  EM 1B
1152 IF YP=0 THEN R=R+1:Q$(R)="/♠\¯<13SPACE>| |T¯
  |_  ♥¯ L¯/| | |":REM 10
1153 IF YP=0 THEN R=R+1:Q$(R)="----------------
  ----------------":RETURN:REM 49
1154 Q$(R)=E$+"<SHIFT+SPACE>"+TY$+" "+TR$+" "+SE$+
  "  "+PI$:REM A5
1156 RETURN:REM AD
1200 IF TR$="" THEN TR$=CHR$(0):REM 55
1202 IF SE$="" THEN SE$=CHR$(0):REM 3B
1204 TR=ASC(TR$):SE=ASC(SE$):REM 8F
1206 IF TR<10 THEN TR$=STR$(TR)+CHR$(32):REM C7
1208 IF SE<10 THEN SE$=STR$(SE)+CHR$(32):REM 9F
1210 IF TR>9 THEN TR$=STR$(TR):REM BA
1212 IF SE>9 THEN SE$=STR$(SE):REM 92
1214 RETURN:REM A8
1300 IF PI$="" THEN PI$=CHR$(0):REM 3C
1302 IF PH$="" THEN PH$=CHR$(0):REM 3C
1304 PI=ASC(PI$):PH=ASC(PH$):REM 76
1306 PI=PI+256*PH:REM C3
1308 PI$=STR$(PI):REM CD
1310 RETURN:REM A5
2000 RA=1:E$="":D=0:LI=0:FOR X=1 TO LEN(Q$(K)):REM
   D9
2002 A$=MID$(Q$(K),X,1):REM A7
2004 IF RA=1 AND A$=CHR$(32) THEN E$=E$+" ":GOTO 2
  010:REM 65
2005 IF A$=" " THEN LI=0:GOTO 2010:REM A3
```

```
2006 IF LI=0 THEN LI=1:D=D+1:FF$(D)="":REM E6
2008 IF LI=1 THEN FF$(D)=FF$(D)+A$:RA=0:REM 1C
2010 NEXTX:REM 5A
2012 IF D<3 THEN 2042:REM FE
2020 R$=E$+FF$(1):FF$(1)=R$:R$="":REM 8D
2022 SM=D:ME=0:REM 20
2024 FOR X=1 TO SM : ME=ME+LEN(FF$(X)):NEXTX:REM D
  8
2028 D=1:S=1:REM 79
2030 FF$(D)=FF$(D)+" ":ME=ME+1:IF ME=RV THEN 2038:
  REM 2E
2032 D=D+S:IF D=SM THEN S=-1:D=SM-1:GOTO 2030:REM
  02
2034 IF D<1 THEN D=1:S=1:GOTO 2030:REM 1D
2036 GOTO 2030:REM C9
2038 Q$(K)="":REM 5F
2040 FOR X=1 TO SM:Q$(K)=Q$(K)+FF$(X):NEXT X:REM A
  4
2042 IF SIGN=1 THEN SIGN=0:Q$(K)="▌"+Q$(K):REM 80
2043 FOR X=1 TO SM:FF$(X)="":NEXTX:REM 9E
2044 RETURN:REM AA
```

REM-killer

Mikko Nykyri

Ohjelma poistaa ohjelmarivin ainoasta REM-lauseesta vain kommenttitekstin jättäen REM-symbolin muistiin. Jos REM sijaitsee samalla rivillä jonkin muun lauseen perässä, poistetaan REM-lause kokonaan. Toisin sanoen REM-lauseella voidaan edelleen varata ohjelmarivi, johon toisaalla viitataan hyppykäskyllä.

Basic-ohjelman alun koodi hakee muistin nollasivulta toimien siten myös muistissa ylemmäksi siirretyillä ohjelmilla. Ohjelmarivit säilyvät REM-lauseita lukuunottamatta muuten ennallaan, joten ennen ja jälkeen käsittelyn otettujen listausten vertailu on helppoa.

Koodin pituus on 551 tavua. Se varaa muistia seuraavasti:

Koodi	$C200	— $C426
Muuttujat	$002B	— $0032,
	$00FB	— $00FE ja
	$C500	— $C514.

Käytettäessä tämä ohjelma ladataan ensin muistiin, sitten rutistettava ohjelma ja sitten annetaan käsky SYS 49664, jolloin muistissa oleva BASIC-ohjelma menettää kommenttinsa.

```
PROGRAM:REM-KILLER

100 REMIT POIS:REM 4D
110 REM M. NYKYRI -85:REM 71
120 REM CBM-64:REM E0
130 I=-1: C=49664: REM ALKUOSOITE:REM 5D
140 I=I+1: READ A:REM 57
150 IF A=-1 THEN PRINT"<DOWN>VALMIS.  KAYNNISTYS S
  YS 49664-KASKYLLA.": END:REM FD
160 POKE C+I,A:REM EA
170 PRINT I,A:REM DB
180 GOTO 140:REM PALUU LUUPIN ALKUUN:REM B9
190 ::REM D4
200 DATA165,43,141,0,197,165,44:REM 22
210 DATA141,1,197,173,0,197,133:REM 20
```

```
220 DATA251,173,1,197,133,252,160:REM 82
230 DATA0,177,251,141,2,197,160:REM 1E
240 DATA1,177,251,141,3,197,173:REM 25
250 DATA2,197,133,253,173,3,197:REM 2C
260 DATA133,254,160,0,177,253,208:REM 86
270 DATA9,160,1,177,253,208,3:REM C2
280 DATA76,174,195,56,173,2,197:REM 3D
290 DATA233,2,141,6,197,173,3:REM C2
300 DATA197,233,0,141,7,197,24:REM F2
310 DATA173,0,197,105,3,141,4:REM B5
320 DATA197,173,1,197,105,0,141:REM 21
330 DATA5,197,24,173,4,197,105:REM FA
340 DATA1,141,4,197,173,5,197:REM C6
350 DATA105,0,141,5,197,173,4:REM BB
360 DATA197,205,6,197,208,11,173:REM 61
370 DATA5,197,205,7,197,208,3:REM CE
380 DATA76,159,195,173,4,197,133:REM 6F
390 DATA251,173,5,197,133,252,160:REM 8E
400 DATA0,177,251,201,143,208,202:REM 76
410 DATA32,229,195,56,173,6,197:REM 35
420 DATA237,4,197,141,16,197,173:REM 62
430 DATA7,197,237,5,197,141,17:REM 01
440 DATA197,173,2,197,141,10,197:REM 61
450 DATA173,3,197,141,11,197,173:REM 5E
460 DATA0,197,141,8,197,173,1:REM C8
470 DATA197,141,9,197,173,8,197:REM 42
480 DATA133,251,173,9,197,133,252:REM 92
490 DATA56,160,0,177,251,237,16:REM 2B
500 DATA197,145,251,160,1,177,251:REM 89
510 DATA237,17,197,145,251,173,10:REM 8B
520 DATA197,133,253,173,11,197,133:REM C0
530 DATA254,160,0,177,253,208,9:REM 28
540 DATA160,1,177,253,208,3,76:REM F6
550 DATA30,195,173,10,197,141,8:REM 29
560 DATA197,173,11,197,141,9,197:REM 6C
570 DATA173,8,197,133,251,173,9:REM 35
580 DATA197,133,252,160,0,177,251:REM 8E
590 DATA141,10,197,160,1,177,251:REM 57
600 DATA141,11,197,76,193,194,24:REM 5F
610 DATA173,4,197,105,1,141,12:REM E9
```

```
620 DATA197,173,5,197,105,0,141:REM 28
630 DATA13,197,24,173,12,197,109:REM 5F
640 DATA16,197,141,14,197,133,251:REM 8F
650 DATA173,13,197,109,17,197,141:REM 96
660 DATA15,197,133,252,24,173,14:REM 5B
670 DATA197,105,1,141,14,197,173:REM 5E
680 DATA15,197,105,0,141,15,197:REM 2A
690 DATA173,12,197,133,253,173,13:REM 90
700 DATA197,133,254,160,0,177,251:REM 8A
710 DATA145,253,173,14,197,197,45:REM 99
720 DATA208,10,173,15,197,197,46:REM 63
730 DATA208,3,76,140,195,24,173:REM 2B
740 DATA12,197,105,1,141,12,197:REM 22
750 DATA173,13,197,105,0,141,13:REM 1E
760 DATA197,76,47,195,169,0,133:REM 3E
770 DATA252,169,45,32,177,195,169:REM A5
780 DATA47,32,177,195,169,49,32:REM 41
790 DATA177,195,173,2,197,141,0:REM 34
800 DATA197,173,3,197,141,1,197:REM 32
810 DATA76,199,195,96,0,0,133:REM D0
820 DATA251,56,160,0,177,251,237:REM 59
830 DATA16,197,145,251,160,1,177:REM 5E
840 DATA251,237,17,197,145,251,96:REM 9C
850 DATA56,173,0,197,237,16,197:REM 3A
860 DATA141,0,197,173,1,197,237:REM 30
870 DATA17,197,141,1,197,169,0:REM 02
880 DATA141,16,197,141,17,197,76:REM 6C
890 DATA10,194,169,0,141,20,197:REM 2B
900 DATA24,173,0,197,105,6,141:REM EF
910 DATA18,197,173,1,197,105,0:REM F9
920 DATA141,19,197,56,173,4,197:REM 39
930 DATA237,18,197,141,18,197,173:REM 9F
940 DATA5,197,237,19,197,141,19:REM 3C
950 DATA197,169,0,237,20,197,48:REM 3C
960 DATA17,56,173,4,197,233,2:REM CE
970 DATA141,4,197,173,5,197,233:REM 36
980 DATA0,141,5,197,96,-1:REM 00
```

Sprite-editori

Antti Raudaskoski

Ohjelma helpottaa spritejen luontia Kuusnelosella. Sen avulla voi suunnitella sekä yksi- että monivärispritejä. Lopuksi ohjelma tekee suunnitelluista spritehahmoista valmiita DATA-lauseita, joita voi käyttää omissa ohjelmissa.

Aluksi ohjelmassa on pieni tauko, jonka aikana kone lataa muistiin konekieli- ja merkkidatat ja tekee muut alkuvalmistelut. Sitten näytölle ilmestyy kenttä, jolla spritet luodaan käyttämällä eri näppäimiä kursorin ohjaukseen, värien vaihtoon jne. Näppäimet löytyvät ohjesivulta, jonka saa esille painamalla F1-näppäintä.

Sprite ja sen suurennokset ilmestyvät ruudun oikeaan ylänurkkaan sitä mukaa kun spriteä luodaan. Ohjelmalla voi suunnitella samanaikaisesti 50 eri spriteä. Spriten saa DATA-lauseiksi siirtämällä sen F2-näppäintä painamalla muistiin ASCII-koodeina. Lopetuksen yhteydessä konekielisen lopetusohjelman on sitten helppo muodostaa DATA-lauseita järjestyksessä olevista ASCII-koodeista. Lopputuloksena muistiin jää vain suunniteltujen spritejen datat.

Luotava sprite on muistilohkossa 13 (muistipaikat 832-895). Konekieliohjelmat, uudelleen määritellyt merkit ja spritet ASCII-koodeina sijaitsevat muistipaikasta 14336 eteenpäin. Spritejen kuvaukset ja lopetusohjelma ovat muistipaikasta 49152 eteenpäin.

Ohjelma koostuu seuraavista osista:

```
  10 —  330 aloitus
 340 —  600 testaa painallus
1000 — 1430 kursorin liikutus
1500 — 1840 pikseli päälle/pois
1990 — 2110 värien vaihto
2200 — 2230 yksiväri/moniväri
2400 — 2720 spritealueen vieritys
2800 — 2820 kursori pois
2900 — 2940 spritealueen täytö/tyhjennys
3000 — 3010 ruudun peito/paljastus
3100 — 3120 merkkigrafiikan värit paikalleen
3500 — 3630 sprite muistiin ASCII-koodeina datarivejä varten
```

3800 — 3840 lopetus ja ASCII-koodit datariveiksi
4000 — 4150 spriten vaihto
4200 — 4220 kursori päälle/pois
5000 — 5150 ohjesivu
10000 — 12360 konekieli- ja merkkidatat

Joitakin muuttujia:

AS — kursorin asema
AL — kursorin alla oleva merkki
X — kursorin X-asema
Y — kursorin Y-asema
SC — spriten väri
TC — tekstin väri
M1 — moniväri 1
M2 — moniväri 2
CU — kursori päällä/pois
Mc — yksiväri/moniväri

```
PROGRAM:SPRITE-EDITORI

10 REM ********************************:REM DD
20 REM ** SPRITE-EDITORI              **:REM 02
30 REM ** (C)  ANTTI RAUDASKOSKI     **:REM 44
40 REM ** SYYSKUU  1985              **:REM 14
50 REM ********************************:REM E1
100 POKE52,56:POKE56,56:CLR:POKE53280,14:POKE53281
  ,6:GOSUB3000:REM 8B
110 FORI=0TO68:READA:POKE16384+I,A:NEXTI:REM E9
120 SYS16384:POKE53272,(PEEK(53272)AND240)+14:REM
  F3
130 FORI=14880TO14903:POKEI,255:NEXTI:REM E6
140 FORI=14904TO14927:READA:POKEI,A:NEXTI:REM 26
145 FORI=16384TO17213:READA:POKEI,A:NEXTI:REM 26
150 FORI=0TO137:READA:POKE53000+I,A:NEXTI:REM 0C
155 FORI=17214TO17311:READA:POKEI,A:NEXTI:REM 1F
160 SYS17214:REM 95
170 POKE17220,PEEK(17217):POKE17221,PEEK(17218):PO
  KE17217,64:POKE17218,3:REM 3E
```

```
200 AS=1105:AL=0:MC=0:X=0:Y=0:H=0:CU=0:MU=0:SC=14:
  TC=14:M1=15:M2=16:T=0:FC=0:REM D3
205 RI=10000:MS=0:SM=0:SL=18432:REM 06
210 POKE53287,14:POKE53288,14:POKE53289,14:POKE532
  90,14:POKE649,1:REM CB
220 POKE2040,13:POKE2041,13:POKE2042,13:POKE2043,1
  3:POKE18432,191:REM 8E
230 POKE53285,15:POKE53286,16:POKE53271,10:POKE532
  77,12:POKE53264,12:REM 55
240 POKE53248,240:POKE53249,66:POKE53250,240:POKE5
  3251,106:REM EB
250 POKE53252,32:POKE53253,66:POKE53254,32:POKE532
  55,106:POKE53276,0:REM 54
255 PRINTCHR$(8):REM B3
260 PRINT"<CLR/HOME>──────────────SPRITE-EDITORI──
  ──────────┐";:REM 7B
270 FORI=0TO23:PRINT"┬";:NEXTI:PRINT"┐";:REM EC
280 FORI=0TO20:PRINT"<DOWN><LEFT>┤";:NEXTI:PRINT"<
  DOWN><LEFT>┘":REM 1B
290 PRINT"<UP>└";:FORI=0TO23:PRINT"┴";:NEXTI:REM 7
  D
300 PRINT"<HOME><DOWN><RIGHT>";:FORI=0TO20:PRINT"<
  DOWN><LEFT>├";:NEXTI:REM 1D
310 PRINTTAB(29);"OHJEET: |<RVS ON>1":REM 36
315 PRINT"<3UP><12LEFT>"MS+1"<LEFT>. SPRITE":SYS17
  228:IFMC=1THENSYS16507:REM 90
320 GOSUB3100:POKE53269,15:REM DD
330 GOSUB3010:GOTO1400:REM 12
340 SYS17214:REM 95
341 GETA$:IFA$=""THEN340:REM 52
343 IFA$="+"THENMS=MS+1:GOTO4000:REM 79
344 IFA$="-"THENMS=MS-1:GOTO4000:REM 7E
350 IFA$="<UP>"THEN1000:REM 8E
360 IFA$="<DOWN>"THEN1100:REM 10
370 IFA$="<RIGHT>"THEN1200:REM 1E
380 IFA$="<LEFT>"THEN1300:REM A0
390 IFA$="A"ORA$="♠"THEN1500:REM 8F
400 IFA$="S"ORA$="♥"THEN1600:REM AC
410 IFA$="W"ORA$="○"THENIFMC=1THEN1700:REM 72
420 IFA$="Z"ORA$="♦"THENIFMC=1THEN1800:REM 7A
```

```
430 IFA$="1"THENGOSUB2000:GOTO340:REM B8
440 IFA$="2"THENGOSUB2030:GOTO340:REM BD
450 IFA$="3"THENGOSUB2060:GOTO340:REM C2
460 IFA$="4"THEN2090:REM 3D
470 IFA$="5"THEN2100:REM 37
480 IFA$="6"THEN2110:REM 3A
490 IFA$="/"THEN2400:REM 36
500 IFA$="@"THEN2500:REM 40
510 IFA$=";"THEN2700:REM 3E
520 IFA$=":"THEN2600:REM 3D
530 IFA$="<F1>"THEN2300:REM 86
540 IFA$="<F2>"THEN3500:REM 8E
550 IFA$="<F3>"THEN2200:REM 88
560 IFA$="<F4>"THEN2900:REM 94
570 IFA$="<F5>"THEN4200:REM 8D
580 IFA$="<F6>"THEN3800:REM 97
600 GOTO340:REM 66
1000 REM YLOS:REM EC
1010 IFPEEK(AS-40)=114THEN340:REM 85
1020 GOSUB2800:AS=AS-40:Y=Y-1:GOTO1400:REM FC
1100 REM ALAS:REM C7
1110 IFPEEK(AS+40)=113THEN340:REM 83
1120 GOSUB2800:AS=AS+40:Y=Y+1:GOTO1400:REM F9
1200 REM OIKEALLE:REM ED
1210 H=1:IFMC=1THENH=2:REM 27
1220 IFPEEK(AS+H)=115THEN340:REM 6B
1230 GOSUB2800:AS=AS+H:X=X+H:H=0:REM AB
1240 GOTO1400:REM C5
1300 REM VASEMMALLE:REM 8F
1305 H=1:IFMC=1THENH=2:REM 2C
1310 IFPEEK(AS-1)=107THEN340:REM 57
1320 GOSUB2800:REM 10
1340 AS=AS-H:X=X-H:H=0:REM 2D
1400 REM KURSORI PAALLE:REM 87
1402 AL=PEEK(AS):REM 9B
1405 IFCU=1THEN340:REM 25
1410 H=230:IFAL=32THENH=102:REM 1F
1420 POKEAS,H:IFMC=1THENPOKEAS+1,H:REM 87
1430 GOTO340:REM 98
1500 REM PIKSELI POIS:REM F6
```

```
1510 H=X:AL=32:POKEAS,102:POKEAS+54272,TC:REM B8
1520 IFMC=1THENPOKEAS+1,102:POKEAS+54273,TC:REM 52
1530 MU=832+Y*3+INT(H/8):POKEMU,PEEK(MU)AND(255-2↑
  (7-(HAND7))):REM 60
1540 IFMC=1ANDH=XTHENH=H+1:GOTO1530:REM 9B
1550 H=0:GOTO340:REM 8A
1600 REM SPRITE-VARI PIKSELI PAALLE:REM A1
1610 AL=70:POKEAS,230:POKEAS+54272,PEEK(53287):REM
   8E
1620 IFMC=1THENPOKEAS+1,230:POKEAS+54273,SC:REM 54
1630 MU=832+Y*3+INT(X/8):POKEMU,PEEK(MU)OR(2↑(7-(X
  AND7))):REM 86
1640 IFMC=1THENMU=832+Y*3+INT((X+1)/8):POKEMU,PEEK
  (MU)AND(255-2↑(7-((X+1)AND7))):REM 98
1650 GOTO340:REM 9C
1700 REM M.1-VARI PIKSELI PAALLE:REM 77
1710 AL=68:POKEAS,230:POKEAS+1,230:POKEAS+54272,M1
  :POKEAS+54273,M1:REM 86
1720 MU=832+Y*3+INT(X/8):POKEMU,PEEK(MU)AND(255-2↑
  (7-(XAND7))):REM 81
1730 MU=832+Y*3+INT((X+1)/8):POKEMU,PEEK(MU)OR(2↑(
  7-((X+1)AND7))):REM E1
1740 GOTO340:REM 9C
1800 REM M.2-VARI PIKSELI PAALLE:REM 79
1810 AL=69:POKEAS,230:POKEAS+1,230:POKEAS+54272,M2
  :POKEAS+54273,M2:REM 8A
1820 MU=832+Y*3+INT(X/8):POKEMU,PEEK(MU)OR(2↑(7-(X
  AND7))):REM 87
1830 MU=832+Y*3+INT((X+1)/8):POKEMU,PEEK(MU)OR(2↑(
  7-((X+1)AND7))):REM E2
1840 GOTO340:REM 9D
1990 REM VARIEN VAIHDOT:REM 8B
2000 SC=(SCAND15)+1:POKE53287,SC:POKE53288,SC:POKE
  53289,SC:POKE53290,SC:REM DE
2010 POKE2,SC:POKE821,70:SYS16451:REM B7
2020 IFAL=70THENPOKEAS+54272,SC:IFMC=1THENPOKEAS+5
  4273,SC:REM 12
2025 RETURN:REM A9
2030 M1=(M1AND15)+1:POKE53285,M1:POKE2,M1:POKE821,
  68:SYS16451:REM 1B
```

```
2040 IFAL=68THENPOKEAS+54272,M1:IFMC=1THENPOKEAS+5
  4273,M1:REM EB
2050 RETURN:REM A7
2060 M2=(M2AND15)+1:POKE53286,M2:POKE2,M2:POKE821,
  69:SYS16451:REM 24
2070 IFAL=69THENPOKEAS+54272,M2:IFMC=1THENPOKEAS+5
  4273,M2:REM F1
2080 RETURN:REM AA
2090 POKE53281,(PEEK(53281)AND15)+1:GOTO340:REM 92
2100 POKE53280,(PEEK(53280)AND15)+1:GOTO340:REM 88
2110 TC=(TCAND15)+1:POKE2,TC:SYS16384:GOTO340:REM
  BC
2200 REM MOODIN VAIHTO:REM 39
2210 GOSUB3000:GOSUB2800:REM 8C
2215 X=0:Y=0:AS=1105:AL=PEEK(AS):REM 6F
2220 IFMC=0THENPOKE53276,15:SYS16507:MC=1:GOSUB301
  0:GOTO1400:REM 73
2230 POKE53276,0:MC=0:SYS16668:GOSUB3100:GOSUB3010
  :GOTO1400:REM 08
2300 REM OHJEET:REM 68
2305 GOSUB3000:REM 0D
2310 POKE53269,0:GOSUB5000:GOSUB3100:GOSUB3010:REM
   D5
2320 GETA$:IFA$<>"<F1>"THEN2320:REM 73
2330 GOSUB3000:GOSUB3100:REM 89
2360 IFAL=32THEN260:REM 50
2370 ON(AL-67)GOSUB2040,2070,2020:REM 0C
2380 GOTO260:REM 9E
2400 REM SPRITEN LIIKUTUS: ALAS:REM A4
2410 GOSUB3000:GOSUB2800:REM 8E
2420 SYS16757:GOSUB3100:GOSUB3010:GOTO1400:REM 05
2500 REM SPRITEN LIIKUTUS: YLOS:REM CB
2510 GOSUB3000:GOSUB2800:REM 8F
2520 SYS16849:GOSUB3100:GOSUB3010:GOTO1400:REM 08
2600 REM SPRITEN LIIKUTUS: VAS.:REM 9D
2610 GOSUB3000:GOSUB2800:H=1:IFMC=1THENH=2:REM 2D
2620 FORI=1TOH:SYS16943:GOSUB3100:NEXTI:H=0:GOSUB3
  010:GOTO1400:REM 78
2700 REM SPRITEN LIIKUTUS: OIK.:REM 97
2710 GOSUB3000:GOSUB2800:H=1:IFMC=1THENH=2:REM 2E
```

```
2720 FORI=1TOH:SYS17049:GOSUB3100:NEXTI:H=0:GOSUB3
  010:GOTO1400:REM 77
2800 REM KURSORI POIS:REM 18
2810 POKEAS,AL:IFMC=1THENPOKEAS+1,AL:REM 15
2820 RETURN:REM AC
2900 REM TAYTA/TYHJENNA:REM C2
2910 GOSUB3000:REM OF
2920 IFFC=0THENPOKE821,0:POKE820,32:FC=1:GOTO2940:
  REM AE
2930 FC=0:POKE821,255:POKE820,70:IFMC=1THENPOKE820
  ,69:REM 80
2940 SYS17153:GOSUB3100:GOSUB3010:GOTO1400:REM 03
3000 POKE53265,PEEK(53265)AND239:RETURN:REM RUUDUN
   PEITTO:REM FF
3010 POKE53265,PEEK(53265)OR16:RETURN:REM RUUDUN P
  ALJASTUS:REM 79
3100 REM MERKKIGRAFIIKAN VARIT:REM 7D
3110 POKE2,SC:POKE821,70:SYS16451:POKE2,M1:POKE821
  ,68:SYS16451:REM D6
3120 POKE2,M2:POKE821,69:SYS16451:POKE2,TC:SYS1638
  4:RETURN:REM 62
3500 REM SPRITE MUISTIIN:REM F5
3510 GOSUB3000:FORI=1TO4:A$(I)="":NEXTI:T=832:SM=S
  M+1:IFSM=51THEN340:REM 84
3512 RI$=STR$(RI):POKE895,0:REM 3B
3515 RI$=RIGHT$(RI$,LEN(RI$)-1):A$(0)=RI$+" REM SP
  RITE"+STR$(SM)+"■":RI=RI+10:REM DA
3520 FORI=1TO4:H=0:REM 2E
3530 T$=STR$(PEEK(T)):T$=RIGHT$(T$,LEN(T$)-1):T=T+
  1:H=H+1:REM 22
3540 A$(I)=A$(I)+T$:IFH<16THENA$(I)=A$(I)+",":GOTO
  3530:REM A1
3550 NEXTI:H=0:REM 44
3580 FORI=1TO4:RI$=STR$(RI):RI$=RIGHT$(RI$,LEN(RI$
  )-1):REM E5
3590 A$(I)=RI$+" DATA "+A$(I)+"■":RI=RI+10:NEXTI:R
  EM A7
3600 A$(4)=A$(4)+"■":I=0:REM F8
3610 FORI=0TO4:REM 3E
3620 FORT=1TOLEN(A$(I)):POKESL,ASC(MID$(A$(I),T,1)
```

```
  ):SL=SL+1:NEXTT:REM AB
3630 NEXTI:SL=SL-1:H=0:GOSUB3010:GOTO340:REM DE
3800 REM LOPETUS JA DATAT RIVEIKSI:REM 3A
3810 GETA$:IFA$=""THEN3810:REM BB
3820 IFA$<>CHR$(13)THEN340:REM B7
3825 POKE53280,14:POKE53281,6:REM 62
3830 GOSUB3000:POKE52,160:POKE56,160:POKE251,0:POK
  E252,72:PRINT"<CLR/HOME>SYS53000":REM C4
3840 POKE53272,21:POKE53269,0:POKE53276,0:POKE631,
  19:POKE632,13:POKE198,2:NEW:REM E4
4000 REM SPRITEN VAIHTO:REM 98
4005 GOSUB3000:REM 0C
4010 IFMS=50THENMS=0:REM D2
4020 IFMS=-1THENMS=49:REM 09
4100 H=49152+64*MS:POKE17217,H-INT(H/256)*256:POKE
  17218,INT(H/256):H=0:REM A8
4110 POKE17220,64:POKE17221,3:SYS17214:SYS17228:IF
  MC=1THENSYS16507:REM B9
4120 POKE17220,PEEK(17217):POKE17221,PEEK(17218):P
  OKE17217,64:POKE17218,3:REM 6D
4140 PRINT"<HOME><19DOWN>"SPC(28);MS+1"<LEFT>. SPR
  ITE ":AL=PEEK(AS):REM BC
4150 SYS17214:GOSUB3100:GOSUB3010:GOTO1400:REM FC
4200 REM KURSORI PAALLE/POIS:REM F2
4205 GOSUB2800:REM 15
4210 IFCU=0THENCU=1:GOTO340:REM 9A
4220 CU=0:GOTO1400:REM 05
5000 PRINT"<CLR/HOME><LT.BLUE>————————————SPRITE-
  EDITORI————————————":REM 57
5010 PRINT"<UP><RVS ON> 1  <RVS OFF> SPRITEV RIN V
  AIHTO":PRINT"<RVS ON> 2  <RVS OFF> MONIV RI 1 VA
  IHTO":REM 6F
5020 PRINT"<RVS ON> 3  <RVS OFF> MONIV RI 2 VAIHTO
  ":PRINT"<RVS ON> 4  <RVS OFF> TAUSTAN V RIN VAIH
  TO":REM 2D
5030 PRINT"<RVS ON> 5  <RVS OFF> REUNAN V RIN VAIH
  TO":PRINT"<RVS ON> 6  <RVS OFF> TEKSTIN V RIN VA
  IHTO":REM E6
5040 PRINT"<RVS ON> @  <RVS OFF> SPRITE YL┐S ":PRI
  NT"<RVS ON> /  <RVS OFF> SPRITE ALAS":REM EC
```

```
5050 PRINT"<RVS ON> :  <RVS OFF> SPRITE VASEMMALLE
  ":PRINT"<RVS ON> ;  <RVS OFF> SPRITE OIKEALLE":R
  EM 3E
5060 PRINT"<RVS ON> A  <RVS OFF> PIKSELI POIS":PRI
  NT"<RVS ON> S  <RVS OFF> SPRITEV RI PIKSELI P L
  LE":REM 2D
5070 PRINT"<RVS ON> W  <RVS OFF> MONIV RI 1 PIKSEL
  I P LLE":REM 83
5090 PRINT"<RVS ON> Z  <RVS OFF> MONIV RI 2 PIKSEL
  I P LLE":REM 89
5095 PRINT"<RVS ON>+JA-<RVS OFF> VAIHDA SPRITE":RE
  M AF
5100 PRINT"<RVS ON>2  <RVS OFF> SPRITE MUISTIIN D
  ATARIVEJ  VARTEN":REM AF
5110 PRINT"<RVS ON>3  <RVS OFF> YKSIV RI/MONIV RI
  ":PRINT"<RVS ON>4  <RVS OFF> SPRITEALUEEN T NTT
  /TYHJENNYS":REM 66
5120 PRINT"<RVS ON>5  <RVS OFF> KURSORI POIS/P L
  LE":PRINT"<RVS ON>CRSR<RVS OFF> KURSORIN LIIKUTU
  S":REM 6F
5130 PRINT"<RVS ON>6  <RVS OFF>+'RETURN' LOPETUS
  JA DATAT RIVEIKSI":REM 1F
5140 PRINT"<RVS ON><4SPACE><RVS OFF> (OHJELMA TUHO
  AA ITSENS )":REM F0
5150 PRINT"<DOWN><11SPACE><RVS ON>1<RVS OFF> PALU
  U EDITORIIN<HOME>":RETURN:REM A4
10000 REM MERKKIJOUKON SIIRTO:REM 48
10010 DATA 173,14,220,41,254,141,14,220,165,1,41,2
  51,133,1:REM FD
10020 DATA 169,0,133,251,133,253,169,208,133,252,1
  69,56,133,254:REM 20
10030 DATA 160,0,177,251,145,253,165,251,201,255,2
  08,4,230,252:REM DB
10040 DATA 230,254,230,251,230,253,165,254,201,64,
  208,230,165,1:REM 07
10050 DATA 9,4,133,1,173,14,220,9,1,141,14,220,96:
  REM 51
10055 REM UUDET MERKIT:REM 32
10060 DATA 255,241,231,195,231,231,231,255:REM 0B
10065 DATA 66,24,60,102,126,102,102,0:REM 04
```

```
10070 DATA 66,60,102,102,102,102,60,0:REM FA
10080 REM TEKSTIN VARIN VAIHTO:REM 4A
10090 DATA 169,0,133,251,133,253,169,4,133,252,169
  ,216,133,254:REM EF
10100 DATA 160,0,177,251,201,68,240,18,201,69,240,
  14,201,70:REM 3D
10110 DATA 240,10,201,230,240,6,165,2,160,0,145,25
  3,165,251:REM 31
10120 DATA 201,255,208,4,230,252,230,254,230,251,2
  30,253,165,251:REM 35
10130 DATA 201,232,208,210,165,252,201,7,208,204,9
  6:REM BC
10140 REM SPRITEN VARIN VAIHTO:REM 4A
10150 DATA 169,81,133,251,133,253,169,4,133,252,16
  9,216,133,254:REM 25
10160 DATA 160,0,177,251,205,53,3,208,6,165,2,160,
  0,145:REM 7D
10170 DATA 253,165,251,201,255,208,4,230,252,230,2
  54,230,251,230:REM 3A
10180 DATA 253,165,251,201,144,208,221,165,252,201
  ,7,208,215,96:REM 19
10185 REM MOODIN VAIHTO (YV-MV):REM 44
10190 DATA 169,81,133,251,133,253,169,4,133,252,16
  9,216,133,254:REM 29
10200 DATA 160,0,177,251,201,32,208,25,200,177,251
  ,201,70,208:REM 9D
10210 DATA 18,173,37,208,133,2,169,68,141,53,3,169
  ,0,240:REM CO
10220 DATA 58,169,0,240,223,160,0,177,251,201,70,2
  08,21,200:REM 3C
10230 DATA 177,251,201,32,208,14,173,39,208,133,2,
  169,70,141:REM 81
10240 DATA 53,3,169,0,240,25,160,0,177,251,201,70,
  208,36:REM BO
10250 DATA 200,177,251,201,70,208,29,173,38,208,13
  3,2,169,69:REM 8E
10260 DATA 141,53,3,165,2,160,0,145,253,200,145,25
  3,173,53:REM 12
10270 DATA 3,160,0,145,251,200,145,251,165,251,201
  ,255,208,4:REM 71
```

```
10280 DATA 230,252,230,254,230,251,230,253,165,251
  ,201,255,208,4:REM 3C
10290 DATA 230,252,230,254,230,251,230,253,165,251
  ,201,145,208,145:REM A1
10300 DATA 165,252,201,7,208,139,96:REM BB
10310 REM MOODIN VAIHTO (MV-YV):REM 3A
10320 DATA 169,81,133,251,169,4,133,252,160,0,177,
  251,201,68:REM 87
10330 DATA 208,15,169,32,160,0,145,251,169,70,200,
  145,251,169:REM B6
10340 DATA 0,240,25,201,70,208,10,169,32,160,1,145
  ,251,169:REM OF
10350 DATA 0,240,11,201,69,208,7,169,70,145,251,20
  0,145,251:REM 47
10360 DATA 165,251,201,255,208,2,230,252,230,251,1
  65,251,201,255:REM 3D
10370 DATA 208,2,230,252,230,251,165,251,201,145,2
  08,182,165,252:REM 43
10380 DATA 201,7,208,176,96:REM 37
10400 REM SPRITEN LIIKUTUS: ALAS:REM D3
10410 DATA 160,59,185,64,3,153,67,3,136,192,255,20
  8,245,169:REM 6C
10420 DATA 0,141,64,3,141,65,3,141,66,3,169,7,133,
  252:REM 20
10430 DATA 160,73,132,251,160,0,132,253,160,40,132
  ,254,164,253:REM D5
10440 DATA 177,251,133,2,169,32,145,251,164,254,16
  5,2,145,251:REM BC
10450 DATA 230,253,230,254,164,253,192,24,208,230,
  162,40,165,251:REM 46
10460 DATA 208,2,198,252,198,251,202,208,245,165,2
  51,201,41,208:REM 21
10470 DATA 203,165,252,201,4,208,197,96:REM 85
10500 REM SPRITEN LIIKUTUS: YLOS:REM FA
10510 DATA 160,0,185,67,3,153,64,3,200,192,63,208,
  245,169:REM F4
10520 DATA 0,141,127,3,141,126,3,141,125,3,169,81,
  133,251:REM DC
10530 DATA 169,4,133,252,160,0,132,253,160,40,132,
  254,164,254:REM AC
```

```
10540 DATA 177,251,133,2,169,32,145,251,164,253,16
  5,2,145,251:REM BC
10550 DATA 230,253,230,254,164,253,192,24,208,230,
  162,40,165,251:REM 47
10560 DATA 201,255,208,2,230,252,230,251,202,208,2
  43,165,251,201:REM 36
10570 DATA 113,208,201,165,252,201,7,208,195,96:RE
  M OC
10800 REM SPRITEN LIIKUTUS: VAS.:REM CE
10810 DATA 169,0,133,2,169,64,133,251,169,3,133,25
  2,160,0:REM EC
10820 DATA 177,251,24,10,145,251,144,14,165,2,240,
  10,198,251:REM 7E
10830 DATA 177,251,9,1,145,251,230,251,200,200,200
  ,192,63,208:REM A7
10840 DATA 227,230,2,230,251,165,251,201,67,208,21
  5,169,81,133:REM E8
10850 DATA 251,169,4,133,252,160,1,177,251,136,145
  ,251,200,169:REM ED
10860 DATA 32,145,251,200,192,24,208,241,162,40,16
  5,251,201,255:REM 11
10870 DATA 208,2,230,252,230,251,202,208,243,165,2
  51,201,153,208:REM 3E
10880 DATA 218,165,252,201,7,208,212,96:REM 87
10900 REM SPRITEN LIIKUTUS: OIK.:REM C8
10910 DATA 169,0,133,2,169,66,133,251,169,3,133,25
  2,160,0:REM EF
10920 DATA 177,251,24,74,145,251,144,12,165,2,240,
  8,200,177:REM 55
10930 DATA 251,9,128,145,251,136,200,200,200,192,6
  3,208,229,230:REM 12
10940 DATA 2,198,251,165,251,201,63,208,217,169,81
  ,133,251,169:REM FC
10950 DATA 4,133,252,160,22,177,251,200,145,251,13
  6,169,32,145:REM E8
10960 DATA 251,136,192,255,208,241,162,40,165,251,
  201,255,208,2:REM 1D
10970 DATA 230,252,230,251,202,208,243,165,251,201
  ,153,208,218,165:REM AA
10980 DATA 252,201,7,208,212,96:REM F9
```

```
11000 REM TAYTA/TYHJENNA:REM E9
11010 DATA 160,0,173,53,3,153,64,3,200,192,63,208,
  248,169:REM EB
11020 DATA 81,133,251,169,4,133,252,160,0,173,52,3
  ,145,251:REM 14
11030 DATA 200,192,24,208,249,162,40,165,251,201,2
  55,208,2,230:REM D9
11040 DATA 252,230,251,202,208,243,165,251,201,153
  ,208,223,165,252:REM 9F
11050 DATA 201,7,208,217,96:REM 2E
12000 REM LOPETUS JA DATAT RIVEIKSI:REM 62
12010 DATA 32,68,229,162,0,160,0,24,32,240,255,177
  ,251,133:REM 15
12020 DATA 2,165,251,201,255,208,2,230,252,230,251
  ,165,2,201:REM 68
12030 DATA 190,240,11,201,191,240,70,32,210,255,16
  9,0,240,225:REM 9C
12040 DATA 169,13,32,210,255,169,83,32,210,255,169
  ,89,32,210:REM 8D
12050 DATA 255,169,83,32,210,255,169,53,32,210,255
  ,169,51,32:REM 90
12060 DATA 210,255,169,48,32,210,255,169,48,32,210
  ,255,169,48:REM CA
12070 DATA 32,210,255,169,19,141,119,2,169,13,141,
  120,2,141:REM 46
12080 DATA 121,2,169,3,133,198,96,173,17,208,9,16,
  141,17:REM CD
12090 DATA 208,169,76,32,210,255,169,105,32,210,25
  5,169,19,141:REM F7
12100 DATA 119,2,169,13,141,120,2,169,2,133,198,96
  :REM 98
12200 REM MUISTIN SIIRTO:REM DC
12210 DATA 160,0,185,0,192,153,64,3,200,192,63,208
  ,245,96:REM EE
12300 REM SPRITE->GRAF.:REM 6A
12310 DATA 169,81,133,251,169,4,133,252,169,64,133
  ,253,169,3:REM 97
12320 DATA 133,254,160,0,132,2,162,0,160,0,177,253
  ,141,52:REM D5
12330 DATA 3,24,14,52,3,176,4,169,32,208,2,169,70,
```

```
  164:REM 2C
12340 DATA 2,145,251,230,2,232,224,8,208,233,230,2
  53,165,253:REM 79
12350 DATA 201,127,208,1,96,165,2,201,24,208,209,1
  62,40,165:REM 4D
12360 DATA 251,201,255,208,2,230,252,230,251,202,2
  08,243,240,188:REM 3E
```

Tarkkuusgrafiikka-piirturi

Jyrki J. J. Kasvi

Tämän ohjelman avulla on mahdollista luoda tarkkuusgrafiikkaruutuja muiden ohjelmien käytettäviksi. Ohjelman avulla luotu kuva talletetaan massamuistilaitteelle, mistä se sitten pienen latausohjelman avulla on helppo ladata muistiin tai jopa tulostaa kirjoittimelle.

Viivaa piirtävää kynää ohjataan joystickillä. Pyyhintätilaan kynä saadaan painamalla tulitusnäppäintä. Samaten siten pääsee takaisin piirtotilaan.

Viivan väri voidaan asettaa painamalla F1:tä tai F3:ta, minkä jälkeen painetaan jotakin numeronäppäintä väliltä 1-9. F1:n painamisen jälkeen viivan väriksi tulee sama kuin normaalisti CONTROL-näppäimen kanssa samaa numeroa painettaessa. F3:n painamisen jälkeen sama kuin COMMODORE-mäppäimen kanssa (CBM+1▶ oranssi).

Sama pätee taustan väriin sillä erolla, että CONTROL+numerolla saatavat värit ovat F5:n takana ja loput F7:n kautta käytettävissä.

Lataus/talletus-valikkoon pääsee painamalla 0-näppäintä. Tällöin voi valita massamuistilaitteen ja tallettaa tai ladata uuden kuvan. Paluu takaisin piirtotilaan ilman toimenpiteitä onnistuu myös.

Ohjelmaa on hyvin helppo laajentaa. Siihen ohjelman osaan, jossa suoritettava toiminto valitaan tulee vain lisätä ehto, jolla testataan, josko käyttäjä haluaa käyttää uutta toimintoa. Tämän jälkeen hypätään kyseisen toiminnon toteuttavaan aliohjelmaan. Tällaisia toimintoja voisivat olla neliöiden ja ympyröiden piirtäminen, alueen värittäminen tai vaikkapa grafiikkaruudulle kirjoittaminen.

PROGRAM: PIIRTURI

```
10 GOTO 9000:REM 63
100 IF(PEEK(B)AND1)=0ANDY>-1THEN Y=Y-1:REM DE
110 IF(PEEK(B)AND2)=0ANDY<201THEN Y=Y+1:REM 11
120 IF(PEEK(B)AND4)=0ANDX>-1THEN X=X-1:REM EO
130 IF(PEEK(B)AND8)=0ANDX<321THEN X=X+1:REM 19
140 IF PEEK(B)=255-16 THEN P=P+1:IF P=2 THEN P=0:R
```

```
  EM 25
200 X1=INT(X/8): Y1=INT(Y/8): AP=YAND7:REM BB
210 MU=8192+X1*8+Y1*320+AP: AP=7-(XAND7): MS=2↑AP:
  REM E3
220 IF P=1 THEN POKE MU,PEEK(MU)ORMS:GOTO240:REM 6
  F
230 POKE MU,PEEK(MU)AND(255-MS):REM 37
240 MU=1024+X1+40*Y1:POKE MU,16*VV+TV:REM 7C
300 POKEV+1,Y+50:REM 89
310 IF X+25>255 THEN POKE V+16,1: POKE V, X+25-256
  : GOTO 330:REM 32
320 POKE V+16,0: POKE V,X+25:REM DD
330 ::REM DO
1000 GET VA$: IF VA$="" THEN 100:REM 21
1010 IF ASC(VA$)<>133THEN 1040:REM 39
1020 GET VV$: IF VV$=""THEN 1020:REM 7F
1025 IF ASC(VV$)<49 OR ASC(VV$)>56 THEN 1020:REM 2
  C
1030 VV=VAL(VV$)-1:REM OF
1040 IF ASC(VA$)<>134THEN 1070:REM 40
1050 GET VV$: IF VV$=""THEN 1050:REM 85
1055 IF ASC(VV$)<49 OR ASC(VV$)>56 THEN 1050:REM 3
  2
1060 VV=VAL(VV$)+7:REM 16
1070 IF ASC(VA$)<>135THEN 1100:REM 3E
1080 GET VV$: IF VV$=""THEN 1080:REM 8B
1085 IF ASC(VV$)<49 OR ASC(VV$)>56 THEN 1080:REM 3
  8
1090 TV=VAL(VV$)-1:REM 13
1100 IF ASC(VA$)<>136THEN 1130:REM 3C
1110 GET VV$: IF VV$=""THEN 1110:REM 7F
1115 IF ASC(VV$)<49 OR ASC(VV$)>56 THEN 1110:REM 2
  C
1120 TV=VAL(VV$)+7:REM 11
1130 IF VA$<>"0" THEN 100:REM BD
1140 FORSI=1024TO2023:POKESI+6144,PEEK(SI):NEXT:RE
  M 6D
1150 POKE 53265, PEEK(53265) AND 223:REM OC
1160 POKE 53272, PEEK(53272) AND 247:REM OF
1170 PRINT"<CLR/HOME><2DOWN>HALUATKO:":REM CO
```

```
1180 PRINT"<DOWN>-<RVS ON>L<RVS OFF>ADATA KUVAN LE
  VYKKEELTA":REM C8
1190 PRINT"-<RVS ON>T<RVS OFF>ALLENTAA KUVAN LEVYK
  KEELLE":REM A3
1200 PRINT"-<RVS ON>P<RVS OFF>ALATA TAKAISIN PIIRT
  OTILAAN?":REM 19
1210 GETGT$:IFGT$=""OR(GT$<>"L"ANDGT$<>"T"ANDGT$<>
  "P")THEN1210:REM 5E
1220 IF GT$<>"L" THEN 1400:REM 11
1230 INPUT"<DOWN>MINKA NIMISESTA TIEDOSTOSTA";NI$:
  REM 11
1240 POKE 53265,PEEK(53265)OR32: POKE 53272, PEEK(
  53272)OR8:REM 92
1250 PRINT"<CLR/HOME>":OPEN 1,8,0,NI$:REM 70
1260 FOR SI=8192 TO 16191:REM 02
1270 GET#1,ME$: IF ME$="" THEN POKE SI,0: GOTO 128
  5:REM 79
1280 POKE SI,ASC(ME$):REM A0
1285 NEXT:REM 0F
1290 FOR SI=1024 TO 2023:REM BD
1300 GET#1,ME$: IF ME$="" THEN POKE SI,0: GOTO 131
  5:REM 6D
1310 POKE SI,ASC(ME$):REM 9A
1315 NEXT:REM 09
1320 GOTO 100:REM 90
1400 IF GT$<>"T" THEN 1600:REM 1B
1410 INPUT"<DOWN>MINKA NIMISEEN TIEDOSTOON";NI$:RE
  M 71
1430 OPEN 1,8,1,NI$:REM D3
1440 FOR SI=8192 TO 16191: PRINT#1,CHR$(PEEK(SI));
  :PRINTSI;:NEXT:REM FF
1450 FOR SI=7168 TO 8167: PRINT#1,CHR$(PEEK(SI));:
  NEXT:REM 38
1460 CLOSE 1:REM 72
1600 FOR SI=1024 TO 2023: POKE SI,PEEK(SI+6144): N
  EXT:REM 6E
1610 POKE 53265,PEEK(53265)OR32: POKE 53272, PEEK(
  53272)OR8:REM 93
1620 GOTO 100:REM 93
8999 REM ALUSTUKSET:REM DC
```

```
9000 V=53248: REM VIC-II -PIIRIN ALKU:REM 46
9010 X=0:Y=0:B=56321:X1=0:X2=0:VV=0:TV=0:P=1:REM 4
  6
9020 PRINT"<CLR/HOME><DOWN>HALUATKO GRAFIIKKAMUIST
  IN TYHJAKSI";:INPUT GT$:REM 76
9030 IF GT$="E" THEN 9070:REM DF
9050 FOR SI=13*64 TO 13*64+62:REM B4
9060 READ SD: POKE SI,SD: NEXT:REM C3
9070 POKE V+21,1:POKE2040,13:REM FF
9080 POKE V+39,8: REM ORANSSI SPRITE:REM 65
9090 PRINT"<CLR/HOME><3DOWN><7RIGHT>TARKKUUSGRAFII
  KKAPIIRTURI":REM AF
9100 PRINT"<3DOWN> F1/F3 + 1-8 - VIIVAN VARIN VAIH
  TAMINEN":REM 5C
9110 PRINT"<DOWN> F5/F7 + 1-8 - TAUSTAN VARIN VAIH
  TAMINEN":REM 96
9120 PRINT"<DOWN> 0 - LATAUS/TALLETUS":REM 72
9130 PRINT"<DOWN> FIRE - PIIRTO/PYYHINTA":REM 7E
9140 PRINT"<DOWN> ILOTIKKU KAKKOSPORTTIIN":REM 59
9150 IF GT$="E" THEN PRINT"<CLR/HOME>":GOTO9240:RE
  M B8
9200 FOR SI=8192 TO 16191: POKE SI,0: NEXT:REM DE
9210 PRINT"<3DOWN>MINKA VARISEN VIIVAN HALUAT (0-1
  5)";: INPUT VV:REM A9
9215 IFVV<00RVV>15THEN9210:REM 64
9220 PRINT"<DOWN>MINKA VARISEN TAUSTAN HALUAT (0-1
  5)";: INPUT TV: V1=TV:REM 81
9225 IFVT<00RVV>15THEN9220:REM 64
9230 FOR SI=1024 TO 2023: POKE SI, 16*VV+TV: NEXT:
  REM 7B
9240 POKE 53265,PEEK(53265)OR32: POKE 53272,PEEK(5
  3272)OR8:REM 9A
9300 GOTO 100:REM 96
10000 REM SPRITE 1:REM DD
10010 DATA 255,128,0,129,192,0,129,32,0,134,144,0,
  132,72,0,156:REM CC
10020 DATA 36,0,146,18,0,225,9,0,208,132,128,72,66
  ,64,36,33:REM 49
10030 DATA 32,18,16,144,9,8,72,4,132,48,2,66,32,1,
  33,64:REM 84
```

```
10040 DATA 0,144,128,0,73,0,0,38,0,0,20,0,0,8,0,0:
  REM 26
60000 REM ********************************:REM 9C
60001 REM *                                *:REM 2F
60002 REM * TARKKUUSGRAFIIKKAPIIRTURI       *:REM AB
60003 REM *                                *:REM 31
60004 REM * WRITTEN BY J. J. J. KASVI       *:REM E0
60005 REM *                                *:REM 33
60006 REM * (C) TECNOPRESS OY AD 1985       *:REM D2
60007 REM *                                *:REM 35
60008 REM ********************************:REM A4
```

CBM-64 MUSIC

Jukka Heino

Keskeytyksellä toteutettua taustamusiikki

Tähän ohjelmistoon kuuluu konekielinen CODE, joka soittaa taustamusiikkia keskeytysperiaatteella, sekä BASIC-kielinen COMPILER, joka kääntää DATA-lauseisiinsa kirjoitettavan musiikkia koskevan tiedon CODEN ymmärtämään muotoon. CODE alkaa muistipaikasta /C000 ja päättyy muistipaikkaan /C230. Levyaseman käyttäjille on mukaan liitetty MUSIC latausohjelma, ja kasettiaseman käyttäjille lataaja käännetyn musiikin lataamista varten. COMPILERIN lopussa on DATA-lauseet yksiäänistä Ukko Noaa varten.

Keskeytysperiaatteen toimintatavasta:

Keskeytys on pääprosessorin (6510) toiminnan kannalta katsoen jokin ulkoinen tapahtuma, joka saa prosessorin keskeyttämään sen, mitä se parhaillaan on tekemässä ja suorittamaan keskeytyksen aiheuttaneen tehtävän ennen paluuta normaaliin päiväjärjestykseen. C64 käyttää kolmentyyppisiä keskeytyksiä, joista tässä käsitellään ns. hardware IRQ:ta (Interrupt ReQuest cueue). On mahdollista "tunkeutua" C64:n toimintaan liittyvien keskeytysten väliin asettamalla muistipaikoissa 788 ja 789 sijaitseviin osoitinvektoreihin oman konekielirutiinin alkuosoite. Ensin osoitteen LO-tavu, sitten HI-tavu. Sitä ennen on kuitenkin kytkettävä keskeytykset pois päältä, jotta keskeytys ei pääsisi alkamaan kesken osoitteen vaihdon. Kun IRQ-tilanne ilmenee, tapahtuu jompikumpi seuraavista: Asetetaan keskeytyslippu päälle ja aletaan suorittaa keskeytyksen aiheuttanutta tapahtumaa vektorin osoittamasta alkuosoitteesta, tai jos keskeytys on jo päällä, asetetaan uusi keskeytys jonoon odottamaan omaa vuoroaan. Keskeytyksien kanssa operoiminen vaatii konekielen käyttämistä, sillä BASIC on siihen liian hidas. Kun musiikkia soittava keskeytysrutiini saadaan tällä tavoin ujutetuksi C64:n omien keskeytysrutiinien joukkoon, se ei toimiessaan millään tavoin häiritse eikä hidasta normaalin BASIC-ohjelman kulkua.

Kun ohjelmat on kirjoitettu muistiin, ne on tallennettava seuraavilla nimillä: Lataaja nimellä "MUSIC", heksadesimaalilistaus nimellä " CODE" ja varsinainen ohjelma nimellä " COMPILER". Huomaa, että " CODE" ja " COMPILER" alkavat välilyönnillä.

Ohjelmiston lataus:

Levyaseman käyttäjät:

LOAD"MUSIC",8 ja RUN

Kasettiaseman käyttäjät:

Latausohjelman rivillä 20 on 8 paikalle vaihdettava 1. Ennen latausta on ohjelmat tallennettava kasetille järjestyksessä: lataaja, " CODE", ja COMPILER. Lisäksi COMPILERIN rivi 1890 on muutettava muotoon:

1890 OPEN 1,1,1,N$:PRINT#1,DA

BASIC-kielinen COMPILER kääntää DATA-lauseisiin kirjoitetun musiikin konekielisen soitto-ohjelman ymmärtämään muotoon. Käytettävien äänigeneraattoreiden lukumäärä on valittavissa, joten on mahdollista jättää esim. yksi generaattori vapaaksi ääniefektejä varten. Ohjelma sallii aaltomuotojen, ADSR:n ja suotimien vapaan käytön.

COMPILERin käyttöohjeet:

Musiikkia koskeva tieto on kirjoitettava COMPILER-ohjelman DATA-lauseisiin seuraavasti:

1. käytössä olevien äänigeneraattorien määrä
2. kappaleen tempo, joka voi olla 30-150 neljäsosanuottia minuutissa.
3. äänen voimakkuus (0-15)
4. aaltomuoto: Mahdollisia aaltomuotoja ovat kolmio-aalto (T), saha-aalto (S), pulssiaalto (P), jonka yhteydessä on aina ilmoitettava pulssin leveys prosentteina (0-100), esim. P20, sekä kohina (N).
5. Seuraavat 4 DATAA ilmoittavat ADSR:n arvot (0-15). Mikäli soitetaan 2 tai 3 äänistä musiikkia, annetaan tässä myös 2. äänen aaltomuoto ja verhokäyrä sekä 3. äänen aaltomuoto ja verhokäyrä.

6. Mahdollisesti käytössä olevan suotimen tyyppi: High pass, ylipäästö (H), Low pass, alipäästö (L), Band pass, kaistanpäästö (B), tai Notch, reunapäästö (N).
7. suotimen rajataajuus Hertzeinä: 30-12000 Hz
8. suotimen resonanssi (0-15)
9. suotimen käyttökytkin ON/OFF jokaiselle käytettävälle generaattorille

Äänen ominaisuuksia koskevien asetusten jälkeen on vuorossa itse musiikin kirjoittaminen:

— Sävelkorkeudet esitetään sävelnimen ja oktaavialan yhdistelmänä, esim A4, jonka oletusarvo oli 440 Hz. Käytettävissä on sävelet C0, C#0, D0, D#0, ...G7, G#7, A7, A#7.
— Tauko merkitään R-kirjaimella
— Aika-arvoista on käytettävissä $^1/_{16}$, $^1/_8$, $^1/_4$, $^1/_2$, $^1/_1$ nuotit, sekä lisäksi pisteelliset puoli-, neljäsosa-, ja kahdeksasosanuotit.

DATA-lauseisiin nuotit merkitään siten, että esim G4/2 tarkoittaa 4. oktaavin G-ääntä, joka on puolinuotin pituinen. Pisteelliset aika-arvot merkitään kirjoittamalla aikaarvon perään piste: C#3/4. tarkoittaa siis 3. oktaavin Cis säveltä, joka on pisteellisen neljäsosan pituinen eli $^1/_4+^1/_8$.

Nuotit merkitään soittojärjestyksessä peräkkäin, mikäli on kyseessä yksiääninen melodia. Jos käytössä on 2 tai 3 ääntä, on kunkin äänen seuraavat nuotit esitettävä peräkkäin. Siis ensin 1. äänen nuotti, sitten 2. ja 3. äänen nuotit, ja taas sama järjestys seuraavien nuottien kohdalla. Jos halutaan muuttaa äänitietoja kesken kappaleen, annetaan DATA-lauseessa kertomerkki (*), jolloin ohjelma ottaa käyttöön uudet nyt annettavat ääniparametrit.

Datat lopetetaan joko END- tai RPT symbolilla. END:illä lopetettu kappale päättyy soituaan kerran, ja RPT:llä lopetettu toistaa itseään jatkuvasti.

Kun kappale on kirjoitettu, on syytä heti tallentaa se levylle tai kasetille ennen kääntämistä. Tallennuksen jälkeen ajetaan COMPILER, jolloin tapahtuu seuraavaa: Kuvaruutuun tulostuu valikko, ja se laskee n. 10 sekuntia nuottien HI- ja LO arvoja.

Seuraavaksi kone kysyy haluatko virittää. Viritysmoodissa ohjelma soittaa neljännen oktaavin A-sävelen ja näyttää samanaikaisesti taajuuden Hertzeinä. Oletusarvona A4:n taajuus on 440 Hz.

1.=viritys
2.=käännös konekielitiedoksi

3.=stop/ohj.jää muistiin

Jos DATOISSA ei ole ollut virheitä, kysyy ohjelma kääntämisen jälkeen halukkuuttasi konekielitiedoston tallentamiseen.

Huom! Käännetty versio ei toimi, ellei CODEA ole ladattu tai DATA-KOODIA ajettu ennen COMPILERIN lataamista.

Keskeytysmusiikki alkaa soida komennolla SYS 49664.

```
PROGRAM:  CODE

C000: 1616010101010202 = 034
C008: 0201010101010101 = 009
C010: CE06C0AD06C0F004 = 3FB
C018: 4C31EAEAA9028D06 = 38F
C020: C0CE00C0AD00C0CD = 488
C028: 08C0D007AD03C08D = 39C
C030: 04D4EAAD00C0D035 = 434
C038: 2020C18D00D42020 = 2A2
C040: C18D01D42020C18D = 3B1
C048: 00C0C9FFD0034CC0 = 467
C050: C1C901D0062030C1 = 372
C058: 4C38C0C900D00620 = 303
C060: E0C14C38C0AE03C0 = 456
C068: E88E04D4EAAD07C0 = 4AC
C070: C901D0034C31EAEA = 3EE
C078: CE01C0AD01C0CD08 = 3D2
C080: C0D007AD04C08D0B = 3A0
C088: D4EAAD01C0D03420 = 450
C090: 20C18D07D42020C1 = 34A
C098: 8D08D42020C18D01 = 2F8
C0A0: C0C9FFD0034CC0C1 = 528
C0A8: C901D0062030C14C = 2FD
C0B0: 8FC0C900D00620E0 = 3EE
C0B8: C14C38C0AE04C0E8 = 45F
C0C0: 8E0BD4AD07C0C902 = 3AC
C0C8: D0034C31EAEACE02 = 3F4
C0D0: C0AD02C0CD08C0D0 = 494
C0D8: 07AD05C08D12D4EA = 3D6
C0E0: AD02C0D0352020C1 = 375
C0E8: 8D0ED42020C18D0F = 30C
C0F0: D42020C18D02C0C9 = 3ED
C0F8: FFD0034CC0C1C901 = 469
C100: D0062030C14CE5C0 = 3D8
C108: C900D00620E0C14C = 3AC
C110: 38C0AE05C0E88E12 = 3F3
C118: D4EAEA4C31EA0000 = 40F
C120: AD47C2EE21C1D003 = 459
C128: EE22C160FFFFFFFF = 62D
C130: EA2020C18D02D420 = 36E
C138: 20C18D03D42020C1 = 346
C140: 8D03C02020C18D05 = 2E3
C148: D42020C18D06D4EA = 426
C150: A901CD07C0F047EA = 45F
C158: 2020C18D09D42020 = 2AB
C160: C18D0AD42020C18D = 3BA
C168: 04C02020C18D0CD4 = 332
C170: 2020C18D0DD4EAA9 = 402
C178: 02CD07C0F020EA20 = 3B0
C180: 20C18D10D42020C1 = 353
C188: 8D11D42020C18D05 = 305
C190: C02020C18D13D420 = 355
C198: 20C18D14D4EAEA20 = 44A
C1A0: 20C18D15D42020C1 = 358
C1A8: 8D16D42020C18D17 = 31C
C1B0: D42020C18D18D4EA = 438
C1B8: EA60FFFFFFFFFFFF = 744
C1C0: A900A2259DFFD3CA = 4A9
C1C8: D0FAA9318D1403A9 = 3F1
```

```
C1D0: EA8D1503EA4C31EA = 3E0   C208: 8D02C08D06C078A9 = 3C3
C1D8: EAEA00FFFFFFFFFF = 6CF   C210: 108D1403A9C08D15 = 2BF
C1E0: A9308D21C1A9C28D = 440   C218: 03A9308D21C1A9C2 = 3B6
C1E8: 22C160FFFFFFFFFF = 63E   C220: 8D22C1A900A2259D = 37D
C1F0: FFFFFFFFFFFFFFFF = 7F8   C228: FFD3CAD0FA586000 = 51E
C1F8: FFFFFFFFFFFFFFFF = 7F8   C230: C7C7C7C7C7C7C7C7 = 638
C200: A9018D00C08D01C0 = 345
```

```
PROGRAM: COMPILER
1 REM MUSIC COMPILER BY JUKKA HEINO 1985 TAMPERE:R
  EM 5A
5 CLR:REM 16
10 PRINT"<CLR/HOME><6DOWN><BLK><5SPACE>PAAMENU:":R
  EM FC
20 PRINT"<2DOWN>    1) CBM-64:N VIRITTAMINEN":REM 3
  A
30 PRINT"<DOWN>    2) KIRJOITTAMASI MUSIIKIN":REM E
  4
40 PRINT"<6SPACE>TULKITSEMINEN KONEKIELELLE ":REM
  95
45 PRINT"<DOWN>    3) POISTUMINEN OHJELMASTA":REM E
  A
50 PRINT"<2DOWN>  ODOTA HETKI, LASKEN NUOTTIEN ARV
  OJA":REM 67
60 IFJH<>1THENGOSUB 100:REM 72
65 POKE198,0:REM 98
70 PRINT"<HOME><17DOWN><6SPACE>VALITSE TOIMINTA<19
  SPACE>":REM E9
75 PRINT"<2DOWN> HUOM! KONEKIELINEN MUSIIKKI ALKAA
  ":REM F5
76 PRINT"<7SPACE>KOMENNOLLA  SYS 49664":REM 3A
80 GET Q$:IFQ$<>"1"ANDQ$<>"2"ANDQ$<>"3"THEN80:REM
  F2
90 ON VAL(Q$) GOTO 500,1000:REM 6A
99 PRINT"<CLR/HOME>";:END:REM 22
100 K=1.059463094:L=0.059604645:REM 48
110 DIM A(96),A$(96),A%(96,2):REM 6E
120 A(57)=440/L:REM E1
130 FORN=56TO0STEP-1:REM DE
140 A(N)=A(N+1)/K:REM 68
```

```
150 NEXT:REM D5
160 FORN=58TO96:REM 88
170 A(N)=A(N-1)*K:REM 68
180 NEXT:REM D8
190 FORN=0TO96:REM 4E
200 A(N)=INT(A(N)+.5):REM 59
210 NEXT:REM D2
220 B$=" C C#D D#E F F#G G#A A#B ":REM 5B
230 FORM=1TO12:REM 3D
240 C$=MID$(B$,M*2,2):REM 22
250 IFRIGHT$(C$,1)=" "THENC$=LEFT$(C$,1):REM 95
260 FORN=0TO7:REM 14
270 D$=RIGHT$(STR$(N),1):REM 4A
280 A$(M+12*N-1)=C$+D$:REM 38
290 NEXT:NEXT:REM 53
300 FORN=0TO96:REM 47
310 IFA(N)>65535THEN340:REM 0F
320 A%(N,0)=INT(A(N)/256):REM 1B
330 A%(N,1)=INT((A(N)/256-A%(N,0))*256+.5):REM 51
340 NEXT:REM D6
345 A%(96,0)=0:A%(96,1)=0:REM B5
350 RETURN:REM 78
390 :150:30:REM 09
500 GOSUB700:S=54272:FORN=STOS+24:POKEN,0:NEXT:REM
   8C
510 POKES+24,15:POKES+5,0:POKES+6,240:REM 92
520 T=A(57):REM 26
530 POKES+4,17:REM 0D
540 POKES,(255ANDT):POKES+1,INT(T/256):REM FB
542 A(57)=T:REM 2A
543 PRINT"<HOME><19DOWN>"TAB(22);INT(A(57)*L*100+.
  5)/100;"<LEFT> HZ<5SPACE>":REM 01
545 IFPEEK(197)<>64THEN545:REM F5
550 GET Q$:IF Q$=""THENPOKES+4,0:GOTO550:REM 27
552 IF T*L>540 THEN T=T-32:REM 72
554 IF T*L<340 THEN T=T+32:REM 6E
560 IF Q$=CHR$(133)THENT=T+4:GOTO530:REM 43
570 IF Q$=CHR$(134)THENT=T-4:GOTO530:REM 47
580 IF Q$=CHR$(135)THENT=T+32:GOTO530:REM 78
590 IF Q$=CHR$(136)THENT=T-32:GOTO530:REM 7C
```

```
600 IF Q$=CHR$(13)THEN650:REM 57
610 GOTO530:REM 68
650 PRINT"<CLR/HOME><2DOWN>  ODOTA HETKI, VIRITAN
  KOKO ASTEIKON":GOSUB 130:REM 36
660 FORN=S TO S+24:POKEN,0:NEXT:REM 74
670 JH=1:GOTO10:REM 71
700 PRINT"<CLR/HOME><2DOWN> CBM-64:N VIRITTAMINEN:
  ":REM E2
710 PRINT"<DOWN> PAINA JOTAKIN NAPPIA, NIIN KUULET
  ":REM E0
720 PRINT" 1 VIIVAISEN A:N.":REM 40
730 PRINT"<2DOWN> F1 NOSTAA A:TA HIUKAN":REM 9A
740 PRINT"<DOWN> F3 LASKEE A:TA HIUKAN":REM 7B
745 PRINT"<DOWN> F5 NOSTAA A:TA ENEMMAN":REM D4
750 PRINT"<DOWN> F7 LASKEE A:TA ENEMMAN":REM C1
760 PRINT"<DOWN> RETURN PALAA MENUUN":REM 96
770 PRINT"<2DOWN> NYKYINEN A:N TAAJUUS:";INT(A(57)
  *L*100+.5)/100;"<LEFT> HZ<4SPACE>":REM D8
780 RETURN:REM 7F
800 PRINT"<CLR/HOME><DOWN> ERROR?":REM D6
810 PRINT"<DOWN> TEMPON OLTAVA 30:N JA 150:N RAJOI
  SSA.":STOP:REM B3
820 PRINT"VIRHE SOUNDIN MÄÄRITTELYSSÄ":REM 3A
830 STOP:REM E1
850 PRINT"VIRHE NUOTISSA":REM 62
860 STOP:REM E4
880 PRINT"LIIAN KORKEA NUOTTI":REM 7E
885 STOP:REM EB
890 PRINT"VIRHEELLINEN AIKA-ARVO":REM 72
895 STOP:REM EC
896 PRINT"MUISTITILA LOPPUNUT":REM 04
897 STOP:REM EE
990 END:REM 79
1000 GOSUB1990:DA=49712:READLK,T:IFT<30ORT>150THEN
  800:REM DO
1005 POKE49159,LK:POKE49160,2:REM 8F
1010 READ Q$:GOSUB1500:T=INT(((60/(T/4)))*30+.5):T
  =INT(T/16+.5)*16:REM 44
1050 READ Q$:IFMID$(Q$,2,1)="#"THENQ=3:GOTO1070:RE
  M 6E
```

```
1052 IF Q$="*"THENREADQ$:GOSUB1500:GOTO1050:REM F0
1054 IF Q$="RPT"THENPOKEDA+2,0:GOTO1800:REM 1D
1056 IF Q$="END"THENPOKEDA+2,255:GOTO1800:REM 6C
1058 IF LEFT$(Q$,1)="R"THENL=96:Q=1:GOTO1110:REM F
  8
1060 Q=2:REM 87
1070 S$=LEFT$(Q$,Q):REM 0E
1080 L=-1:FORN=0TO96:IFS$=A$(N)THENL=N:REM 25
1085 NEXT:REM 0D
1090 IFL=-1THENSTOP: 850:REM 8C
1100 IFA%(L,0)=0ANDA%(L,1)=0THEN880:REM 8C
1110 POKEDA,A%(L,1):POKEDA+1,A%(L,0):DA=DA+2:REM B
  6
1120 AI$=RIGHT$(Q$,LEN(Q$)-(Q+1)):REM 13
1130 IFAI$="16."ORAI$="1."THEN890:REM 17
1140 AI=VAL(AI$):REM 6F
1150 AI=T/AI:REM 9B
1160 IF RIGHT$(AI$,1)="."THENAI=AI+(AI/2):REM EB
1170 POKEDA,AI:DA=DA+1:REM 10
1175 IF DA>53240 THEN896:REM F4
1180 GOTO1050:REM C9
1500 V=VAL(Q$):REM 02
1505 POKEDA,0:POKEDA+1,0:POKEDA+2,1:DA=DA+3:REM 08
1510 FOR N=1 TO LK:REM A4
1520 READ Q$:W=64:REM 91
1522 IF Q$="N"THEN W=128:GOTO1540:REM 38
1524 IF Q$="S"THEN W=32:GOTO1540:REM 09
1526 IF Q$="T"THEN W=16:GOTO1540:REM 0E
1528 IF LEFT$(Q$,1)<>"P"THEN 820:REM A8
1530 PW=INT(VAL(RIGHT$(Q$,(LEN(Q$)-1)))*40.95):REM
   2F
1535 IF PW>4095 OR PW<0 THEN 820:REM 91
1540 READ AT,DE,SU,RE:REM C7
1545 IF AT*16+DE>255THEN820:REM DB
1546 IF AT*16+DE<0  THEN820:REM 6E
1547 IF SU*16+RE>255THEN820:REM FE
1548 IF SU*16+RE<0  THEN820:REM 91
1550 POKEDA,(255 AND PW):DA=DA+1:REM EF
1552 POKEDA,INT(PW/256):DA=DA+1:REM 39
1554 POKEDA,W:DA=DA+1:REM E3
1556 POKEDA,AT*16+DE:DA=DA+1:REM 68
1558 POKEDA,SU*16+RE:DA=DA+1:REM 8B
```

```
1560 NEXT:REM 0B
1570 READ Q$:REM 5E
1572 IF Q$="L"THENV=V+16:GOTO1580:REM 8B
1574 IF Q$="H"THENV=V+64:GOTO1580:REM 8C
1576 IF Q$="B"THENV=V+32:GOTO1580:REM 83
1578 IF Q$="N"THENV=V+16+64:GOTO1580:REM 28
1579 GOTO820:REM A9
1580 READ CO:CO=INT(CO/5.84472+30.5):REM DF
1585 IF CO>2048 OR CO<0 THEN 820:REM 68
1590 READ RES:IF RES>15 OR RES<0THEN820:REM EC
1595 RES=RES*16:REM 76
1600 FORN=1TOLK:REM A4
1610 READ Q$:IFQ$="ON"THENQ=1:GOTO1630:REM E0
1620 IFQ$="OFF"THENQ=0:GOTO1630:REM 53
1625 GOTO820:REM A1
1630 RES=RES+Q*2↑(N-1):REM 0E
1640 NEXT:REM 0A
1650 POKEDA,(CO AND 8):DA=DA+1:REM 77
1652 POKEDA,INT(CO/8):DA=DA+1:REM CO
1654 POKEDA,RES:DA=DA+1:REM 77
1656 POKEDA,V:DA=DA+1:REM E5
1680 RETURN:REM AF
1800 POKEDA,0:POKEDA+1,0:DA=DA+5:REM 60
1810 IF DA>53240 THEN896:REM F0
1820 PRINT"<CLR/HOME><DOWN> VALMISTA ON.":REM 6C
1830 PRINT"<2DOWN> HALUATKO TALLENTAA KONEKIELISEN
  ":REM 2F
1840 PRINT"<DOWN> VERSION (K/E)":REM E5
1845 POKE198,0:REM FF
1850 GETA$:IFA$<>"K"ANDA$<>"E"THEN1850:REM 82
1860 IF A$="E"THEN1950:REM 87
1886 INPUT"<DOWN> ANNA KAPPALEEN NIMI ";N$:REM 45
1890 OPEN1,8,2,N$+",P,W":PRINT#1,CHR$(00);CHR$(192
  );:REM 60
1900 FORN=49152TODA:PRINT#1,CHR$(PEEK(N));:NEXT:RE
  M 7A
1910 CLOSE1:REM 72
1930 PRINT"<DOWN> VALMIS":FORN=1TO500:NEXT:REM 09
1950 JH=1:GOTO10:REM A3
1990 PRINT"<CLR/HOME><3DOWN>  TULKITSEN DATOJA KON
  EKIELELLE,":REM 80
1992 PRINT"<DOWN>  ODOTA HETKINEN.":REM B2
```

```
1995 RETURN:REM B8
1999 REM DATOILLA YKSIÄÄNINEN UKKO NOOA:REM DF
2000 DATA 1,120,10,P50,2,10,6,3,L,6000,10,OFF:REM
  E4
2010 DATA C4/4,C4/4,C4/4,E4/4:REM CB
2020 DATA D4/4,D4/4,D4/4,F4/4:REM DO
2030 DATA E4/4,E4/4,D4/4,D4/4:REM D1
2040 DATA C4/2,R/2:REM 97
2050 DATA *,10,P20,0,8,4,8,L,6000,10,OFF:REM F8
2060 DATA E5/4,E5/4,E5/4,E5/4:REM DA
2070 DATA G5/2,F5/2:REM C8
2080 DATA D5/4,D5/4,D5/4,D5/4:REM D8
2090 DATA F5/2,E5/2:REM C8
2100 DATA *,15,P20,4,2,5,2,L,6000,10,ON:REM B4
2110 DATA C4/4,C4/4,C4/4,E4/4:REM CC
2120 DATA D4/4,D4/4,D4/4,F4/4:REM D1
2130 DATA E4/4,E4/4,D4/4,D4/4:REM D2
2140 DATA C4/2,R/2,END:REM 9B
```

```
PROGRAM:MUSIC

10 PRINT"<CLR/HOME><WHT>LOADING CODE...":IF A=1THE
  N30:REM 77
20 A=1:LOAD" CODE",8,1:REM 8B
30 PRINT"<CLR/HOME>LOADING COMPILER...":REM AA
40 PRINT"<BLU>LOAD "CHR$(34)" COMPILER"CHR$(34)",8
  ":REM 2D
50 POKE631,19:POKE632,141:POKE633,131:POKE198,3:NE
  W:REM DE
```

```
PROGRAM:LATAUS KASETILTA

1 REM LATAAJA KASETTIA VARTEN:REM 29
10 INPUT"<CLR/HOME><DOWN> ANNA TIEDOSTON NIMI";N$:
  REM 8A
20 OPEN1,1,0,N$:INPUT#1,DA:REM EB
30 FORN=49152TODA:GET#1,A$:REM 01
40 IFA$=""THENA=0:GOTO60:REM 8F
50 A=ASC(A$):REM 70
60 POKEN,A:NEXT:REM C9
70 CLOSE1:REM OE
```

Liite 1
Ohjelmien syöttöohjeet

Yleiset syöttöohjeet

Ennen kuin alat kirjoittaa ohjelmia koneeseesi, tutustu huolella sen ohjekirjaan ja perehdy tarkasti mikrosi ominaisuuksiin. Opettele käyttämään näppäimistöä ja varsinkin editointinäppäimiä — kursorinäppäimet, delete ja insert.

Opette kuinka ohjelmat talletetaan ja ladataan massamuistista. Talleta aina kirjoittamasi ohjelmat ennen koeajoa. Virheellinen POKE-käskyjä sisältävä ohjelma todennäköisesti sekoittaa koko koneen.

Koska useat Kuusnelosen ohjelmat sisältävät vaikealukuisia ja vaikeasti kirjoitettavia erikoismerkkejä, olemme kehittäneet ohjelman, joka muuttaa nämä luettavaan muotoon. se kirjoittaa erikoismerkkien tilalle listaukseen merkin nimityksen lyhennetyssä muodossa. Nimitys on usein sama kuin se, joka löytyy näppäimestä.

Lyhennykset on sijoitettu listauksissa kulmasulkujen väliin. Esimerkiksi käänteinen R-kirjain, reverse on, oli CONTROL+9 kirjoitetaan listauksissa <RVSON> ja viisi kursorin siirtoa vasemmalle <5LEFT>.

Koska välilyönnit ovat tulostuksen muotoilussa tärkeitä, ohjelmamme listaa ne muodossa <6SPACE> (kuusi välilyöntiä). Oheinen taulukko auttaa ymmärtämään tällä ohjelmalla listattuja ohjelmalistauksia.

Tarkistusohjelma

Edellisen ohjelman lisäksi olemme kehittäneet Tarkastaja-ohjelman. Kirjoita Tarkastaja Kuusneloseesi ja tallenna se massamuistilaitteelle tulevaa käyttöä varten.

Basic-kielinen lataaja laskee DATA-lauseissa olevan summan. Mikäli ohjelma ilmoittaa 'VIRHE DATOISSA' olet tehnyt pienen virheen DATA-lauseissa. Tarkasta ne huolella ja korjaa virhe.

Ennen kuin aloitat uuden ohjelman kirjoittamisen tästä kirjasta lataa Tarkastaja muistiin ja käynnistä se komennolla RUN. Ohjelman Basic-kielinen osuus sijoittaa muistiin konekielisen ohjelman, joka laskee kustakin syötettävästä ohjelmarivistä tarkistussumman.

Kun näytölle ilmestyy teksti 'TARKASTAJA OK.' on oikolukuohjelma käyttövalmis. Kirjoita vielä NEW, joka poistaa Tarkastajan Basic-osuuden muistista ja tekee tilaa varsinaiselle tarkastettavalle ohjelmalle:

Kun Tarkastaja on toiminnassa, voit aloittaa ohjelman kirjoittamisen.

Kun ohjelmarivin lopussa painetaan RETURNia, kirjoittaa Tarkastaja kuvaruudun yläreunaan kaksi käänteisenä tulostettua merkkiä, rivin tarkistussumman heksadesimaalimuodossa. Syöttämällä koneelle erilaisia rivejä voit kokeilla Tarkastajan toimintaa.

Listausohjelmamme kirjoittaa tarkistussumman REM-lauseeseen kunkin rivin loppuun. Sinun tarvitsee vain verrata tätä Tarkastajan laskemaan summaan. Mikäli ne ovat samat, on ohjelmarivi kirjoitettu oikein.

Rivien lopussa olevaa heksadesimaalilukua, REM-käskyä ja sitä edellyttävää kaksoispistettä EI SAA kirjoittaa. Ne on vain lisätty helpottamaan ohjelman kirjoittamista. Mikäli ne kirjoittaa, laskee Tarkastaja ne mukaan ja tuloksena on virheellinen tarkistussumma.

Ennen ohjelman ajoa on painettava yht'aikaa RUN/STOP- ja RESTORE-näppäimiä, jolloin Tarkastaja kytkeytyy pois päältä, eikä häiritse ohjelman suoritusta. Se voidaan käynnistää uudelleen käskyllä SYS 679.

Huomaa, että Tarkastaja ei laske välilyöntejä, joten niiden kirjoittamisella ei ole väliä (paitsi PRINT-lauseissa!).

Tarkastaja ei kykene löytämään kahden merkin vaihtumisesta johtuvaa virhettä. Tarkistussumma kun on tällöin aivan sama.

Tarkastaja ei tunnista esimerkiksi ?-merkkiä PRINT käskyksi, eli listauksen käskyjä ei voi lyhennellä.

```
0 FOR A=820 TO 827:READ B:POKEA,B:C=C+B:NEXT
1 FOR A=679 TO 750:READ B:POKEA,B:C=C+B:NEXT
2 IF C<>8484 THEN PRINT"VIRHE DATOISSA.":END
3 SYS 679: PRINT "(CLR)TARKASTAJA OK.": END
5 DATA 201,186,144,3,56,233,57,96
6 DATA 169,2,141,37,3,169,178,141,36
7 DATA 3,96,32,87,241,133,252,8,201
8 DATA 32,240,47,201,13,208,38,165,253
9 DATA 41,15,9,176,32,52,3,141,42
10 DATA 4,165,253,74,74,74,74,9,176
11 DATA 32,52,3,141,41,4,169,0,141
```

```
12 DATA 41,216,141,42,216,169,0,240,3
13 DATA 24,101,253,133,253,165,252,40,96
```

Hextyper V1.3 käyttöohjeet

Hextyper ohjelmaa käytetään konekielilistausten sisäänsyöttöön. Kun olet näppäillyt sen sisään Kuusneloseesi, tallenna se massamuistilaitteelle ENNEN käyttöä. Sen jälkeen kun haluat syöttää jonkin konekieliohjelman muistiin, toimi seuraavasti:

1. Lataa Hextyper V1.3 sisään koneeseesi.
2. Aja ohjelma kirjoittamalla RUN.
3. Paina A aloittaaksesi uuden ohjelman kirjoituksen.
4. Anna koneelle alkuosoite heksadesimaalimuodossa.
5. Nyt kone kirjoittaa rivin alkuun sen rivin alkuosoitteen, jolla kulloinkin ollaan. Jos alkuosoite ei lisäänny rivin kirjoitettuasi, on rivissä kirjoitusvirhe. Rivit ovat konekieliohjelmien listausten heksadesimaalilukurivejä.
6. Halutessasi tallentaa ohjelman, paina rivin alussa pelkkä return, jotta pääsisit menuun. Täällä painat T tallettaaksesi ohjelman.

Kun haluat jatkaa ohjelman kirjoittamista tee seuraavasti:

1. Lataa hextyper V1.3.
2. Aja se.
3. Paina L, ja anna tarvittavat tiedot.
4. Jatka ylläolevasta kohdasta 5.

Kun haluat ajaa ohjelman tee seuraavasti:

1. Talleta ohjelma Hextyperillä syötön jälkeen massamuistilaitteelle.
2. Sammuta virta Kuusnelosestasi.
3. Lataa konekieliohjelma muistiin aivan normaalisti ja aja se joko SYS- tai RUN-käskyllä. Joissakin tapauksissa lataus voidaan myös suorittaa sekundääriseen muistiosoitteeseen (,8,1 tai ,1,1), jolloin ohjelma latauksen jälkeen käynnistyy automaattisesti.

PROGRAM:HEXTYPER V1.3

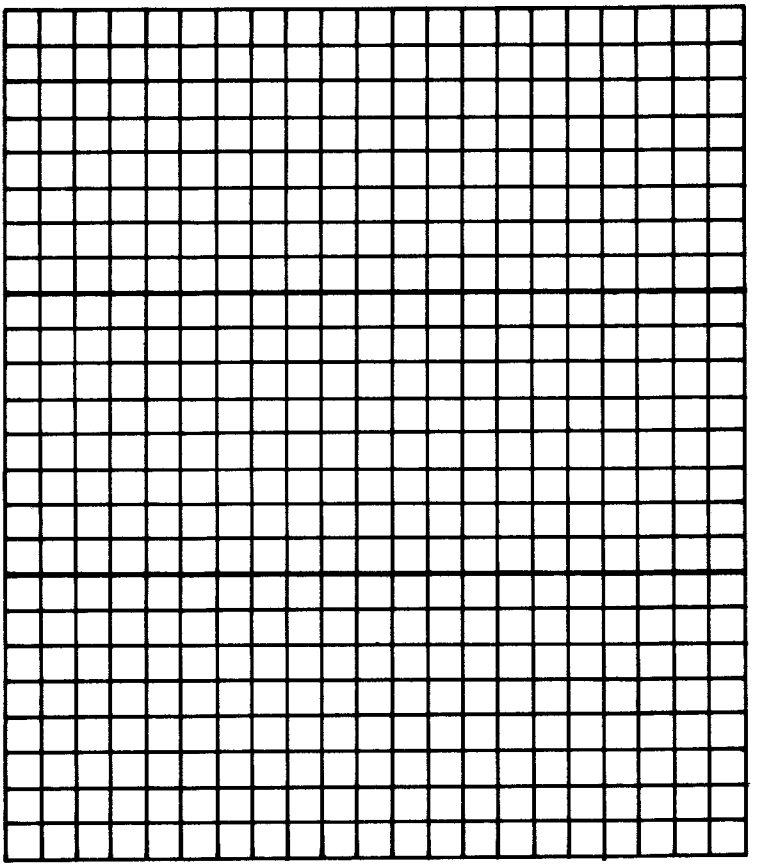
```
60 POKE56,31:POKE55,210:CLR:GOSUB950:D$=CHR$(20):P
  OKE53281,13:POKE53280,6:REM EC
100 PRINT"<CLR/HOME><BLU><RVS OFF> <RVS ON> HEXTYP
  ER 64 V1.3  (C) PETRI HELENIUS ":REM F8
110 PRINT"<2DOWN><2RIGHT>PAINA: <L> LADATAKSESI":R
  EM 4E
120 PRINT"<9RIGHT><T> TALLETTAAKSESI":REM 52
130 PRINT"<9RIGHT><A> ALOITTAAKSESI UUDEN":REM 74
135 PRINT"<13RIGHT>OHJELMAN SYOTTAMISEN":REM 91
136 PRINT"<9RIGHT><J> JATKAAKSESI ASKEISTA<20SPACE
  >SYOTTAMISTA":REM 16
140 PRINT"<9RIGHT><P> POISTUAKSESI OHJELMASTA":REM
   C1
160 GETX$:REM F3
170 IFX$="A"THEN1000:REM 55
180 IFX$="J"THEN1100:REM 60
190 IFX$="P"THEN450:REM 3E
200 IFX$="L"THEN1200:REM 5C
210 IFX$="T"THEN1300:REM 66
290 GOTO160:REM 6B
300 POKE147,0:POKE10,0:POKE186,1:REM 45
310 SYS63276:REM 9B
320 GOSUB700:REM AC
330 IFER=1THENER=0:GOTO400:REM 64
340 POKE829,0:POKE830,32:REM 5A
350 POKE831,(E-S)-((INT((E-S)/256))*256):POKE832,(
  E-S)/256+32:REM FD
360 POKE781,3:REM C7
365 PRINT"<DOWN>OHJELMA LOYDETTY. PAINA VALIA.":RE
  M 30
370 GETA$:IFA$<>" "THEN370:REM 94
380 SYS62813:REM 9E
390 RETURN:REM 7C
400 PRINT"PAINA <J> JATKAAKSESI TAI<21SPACE><L> LO
  PETTAAKSESI.":REM 68
410 GETX$:REM F1
420 IFX$="J"THEN310:REM 2F
430 IFX$="L"THEN100:REM 2F
440 GOTO410:REM 66
450 PRINT"<3DOWN>OLETKO VARMA(K/E)?":REM 31
```

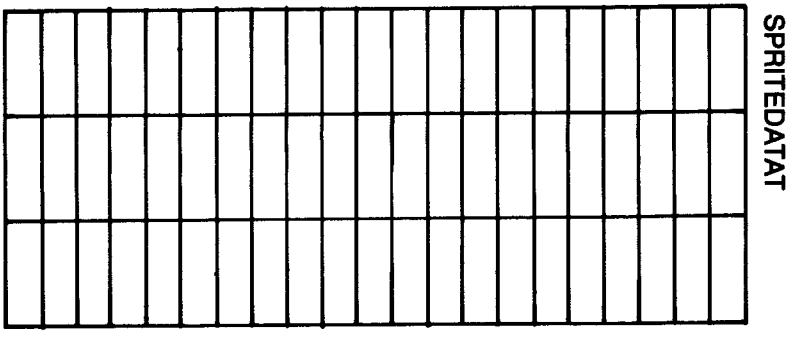
```
460 GETX$:IFX$="E"THEN100:REM C1
470 IFX$<>"K"THEN460:REM 78
480 SYS64738:REM A7
500 INPUT"TIEDOSTONIMI";NI$:REM F7
503 POKE147,0:POKE10,0:POKE186,1:REM 4A
505 POKE780,3:REM C7
510 POKE187,211:POKE188,31:POKE183,LEN(NI$):REM DB
511 NI$=LEFT$(NI$,16):REM 7D
512 FORX=1TOLEN(NI$):POKEX+8146,ASC(MID$(NI$,X,1))
  :NEXT:REM 4A
520 POKE193,S-(INT(S/256)*256):REM 82
530 POKE194,INT(S/256):REM EC
540 POKE174,E-(INT(E/256)*256):REM 67
550 POKE175,INT(E/256):REM DF
557 PRINT"<DOWN><6RIGHT>";:REM 6C
560 SYS63338:REM A1
570 POKE194,32:POKE193,0:REM 5C
580 POKE174,(E-S)-((INT((E-S)/256))*256):REM 5E
590 POKE175,32+INT((E-S)/256):REM 44
595 PRINT"<UP><6RIGHT>";:REM EE
600 SYS63098:RETURN:REM B9
700 FO$="":FORA=833TO848:FO$=FO$+CHR$(PEEK(A)):NEX
  T:REM EB
701 NI$=LEFT$(NI$+"<16SPACE>",16):REM ED
702 FO$=LEFT$(FO$+"<16SPACE>",16):REM EA
705 IFNI$<>FO$THENPRINT"VIRHEELLINEN TIEDOSTONIMI.
  ":ER=1:RETURN:REM BD
710 TY=PEEK(828):REM 9A
720 IFTY<>3THENPRINT"VIRHEELLINEN TIEDOSTOTYYPPI."
  :ER=1:RETURN:REM E8
730 S=PEEK(829)+PEEK(830)*256:E=PEEK(831)+PEEK(832
  )*256:REM 19
740 RETURN:REM 7B
800 IFHX$=""THENRETURN:REM 7B
805 LE=LEN(HX$):DE=0:FORX=LE TO1STEP-1:REM 44
810 AS=ASC(MID$(HX$,X,1)):IF(AS<48ORAS>57)AND(AS<6
  5ORAS>70)THENRETURN:REM 03
820 IFAS>57THENAS=AS-7:REM 5F
830 DE=DE+16↑(LE-X)*(AS-48):REM E9
840 NEXT:RETURN:REM F5
850 HX$="":FORX=4TO1STEP-1:REM 3A
860 NO=INT(DE/16↑(X-1)):DE=DE-NO*16↑(X-1):IFNO>9TH
```

```
  ENNO=NO+7:REM 66
870 HX$=HX$+CHR$(NO+48):NEXT:RETURN:REM A8
950 PRINT"<RVS OFF>";:CU=(CU+1)AND1:IFCUTHENPRINT"
  <RVS ON>";:REM 42
960 PRINT" <LEFT>";:REM 48
970 FORZ=1TO100:NEXT:REM FC
980 RETURN:REM 81
1000 PRINT"<4DOWN>OLETKO VARMA(K/E)?":REM 6A
1010 GETX$:IFX$="E"THEN100:REM E9
1020 IFX$<>"K"THEN1010:REM C8
1030 OH=1:INPUT"<3DOWN>ALKUOSOITE(HEX)";AL$:REM 2C
1040 HX$=AL$:GOSUB800:AL=DE:REM 56
1050 PRINT"<3DOWN>":REM CA
1060 OS=AL:REM 33
1090 LU=1:REM D9
1100 DE=OS:GOSUB850:PRINTHX$;":  ";:REM 00
1110 GETX$:IFX$=""THENGOSUB950:GOTO1110:REM 68
1111 IFX$=CHR$(13)THEN100:REM 82
1112 X=ASC(X$):IF(X<48ORX>57)AND(X<65ORX>70)ANDX<>
  20THEN1110:REM 75
1113 IFX=20ANDLU>1THENLU=LU-1:X$=LEFT$(IN$(LU),1):
  PRINT"<RVS OFF>"D$D$;:GOTO1120:REM B7
1114 IFX=20THEN1110:REM 3F
1115 PRINT"<RVS OFF>";X$;:REM 1D
1120 GETY$:IFY$=""THENGOSUB950:GOTO1120:REM 6C
1121 X=ASC(Y$):IF(X<48ORX>57)AND(X<65ORX>70)ANDX<>
  20THEN1120:REM 77
1122 IFX=20THENPRINT"<RVS OFF>"D$;:GOTO1110:REM B7
1125 PRINT"<RVS OFF>";Y$;:IFLU=9ANDZ$=""THENZ$=X$:
  X$=Y$:GOTO1120:REM DE
1130 IN$(LU)=Z$+X$+Y$:Z$="":LU=LU+1:IFLU=10THEN115
  0:REM C8
1140 PRINT" ";:GOTO1110:REM 08
1150 CH=0:FORY=1TO9:HX$=IN$(Y):GOSUB800:CH=CH+DE:I
  N(Y)=DE:NEXT:REM 70
1160 CH=(CH-IN(9)):REM BA
1170 IFCH<>IN(9)THENPRINT:GOTO1090:REM 77
1180 FORX=0TO7:POKE8192+OS-AL+X,IN(X+1):NEXT:REM D
  8
1190 OS=OS+8:PRINT:GOTO1090:REM B3
1200 REM LOAD:REM C7
1201 PRINT"<2DOWN><K>ASETTI VAI <L>EVY?":REM 1F
1202 GETX$:IFX$="L"THEN1600:REM 29
```

```
1202 GETX$:IFX$="L"THEN1600:REM 29
1203 IFX$<>"K"THEN1202:REM CE
1205 INPUT"TIEDOSTONIMI";NI$:REM 2A
1210 GOSUB300:REM D7
1220 OS=E:AL=S:GOTO1090:REM 7D
1300 REM SAVE:REM D7
1301 PRINT"<2DOWN><K>ASETTI VAI <L>EVY?":REM 20
1302 GETX$:IFX$="L"THEN1700:REM 2B
1303 IFX$<>"K"THEN1302:REM DO
1310 S=AL:E=OS:GOSUB500:GOTO100:REM 93
1600 INPUT"TIEDOSTONIMI";NI$:REM 29
1610 OPEN2,8,2,NI$+",P,R":REM 3E
1620 GET#2,LO$,HI$:REM CA
1630 AL=ASC(LO$+CHR$(0))+256*ASC(HI$+CHR$(0)):REM
  A4
1640 X=8191:REM 33
1650 GET#2,X$:IFST=66THEN1670:REM BF
1660 X=X+1:POKEX,ASC(X$+CHR$(0)):GOTO1650:REM 93
1670 CLOSE2:OS=AL+X-8192:GOTO1090:REM DD
1700 INPUT"TIEDOSTONIMI";NI$:REM 2A
1710 OPEN2,8,2,"@0:"+NI$+",P,W":REM 5D
1720 PRINT#2,CHR$(ALAND255)CHR$(AL/256);:REM 0C
1730 FORX=8192TO8192+OS-AL:REM 19
1740 PRINT#2,CHR$(PEEK(X));:NEXT:REM AE
1750 CLOSE2:GOTO100:REM 79
```

Spriten suunnittelualusta (voidaan valokopioida).



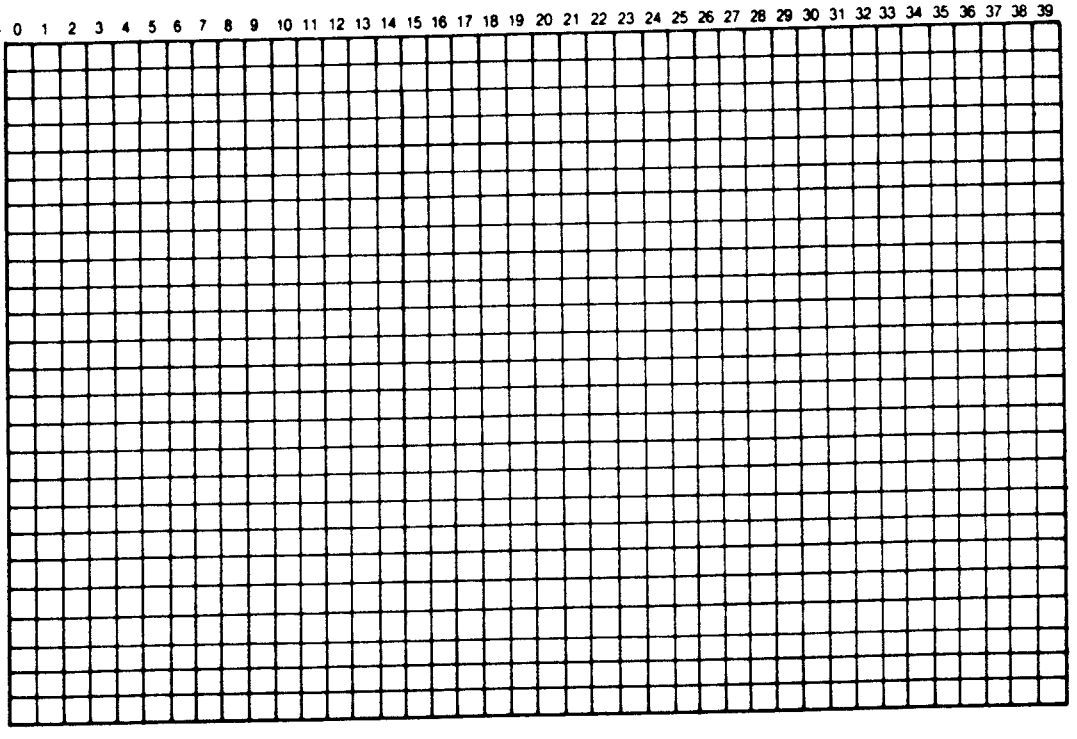


SPRITEDATAT

Rivin ensimmäisten paikkojen osoitteet

Näyttömuistikartta taustan suunnitteluun (voidaan valokopioida)

Värit	Merkit +
55296	1024
55336	1064
55376	1104
55416	1144
55456	1184
55496	1224
55536	1264
55576	1304
55616	1344
55656	1384
55696	1424
55736	1464
55776	1504
55816	1544
55856	1584
55896	1624
55936	1664
55976	1704
56016	1744
56056	1784
56096	1824
56136	1864
56176	1904
56216	1944
56256	1984



Commodore 64 on eräs monipuolisimmista kotitietokoneista. Sen grafiikkaominaisuudet spriteineen ja ääniominaisuudet ohjelmoitavine verhokäyrineen antavat peliohjelmoijalle vahvat työkalut laadukkaiden pelien tekemiseen.

Huvia ja hyötyä Commodore 64 on kirja, joka menee pitemmälle kuin käyttöoppaat. Se antaa tarpeellista tietoa omien ohjelmien rakentamiseen, vinkkejä siitä, millä tavalla ohjelman eri osat käytännössä kootaan toimivaksi kokonaisuudeksi. Ensi sijaisena kohteena on peliohjelmointi, joka kiinnostaa jokaista kotitietokoneen käyttäjää, mutta kirja antaa hyvän perustan muullekin ohjelmoinnille. Lisäksi kirjassa on 16 valmista ohjelmaa, niin pelejä kuin hyötyohjelmiakin, jotka ovat suoraan syötettävissä omaan koneeseen.

Tecnopress Oy